AIR FORCE REPORT NO.
SAMSO-TR-72-181

AEROSPACE REPORT NO.
TR-0172(S2816-75)-1

AD 749482

Aerospace Synthesized Aerodynamic Heating Program

D D C
OCT 11 1972
C

Prepared by M. J. Gyetvay and J. M. Kohlenberger
Technology Division and Mathematics and Computation Center

72 MAR 15

San Bernardino Operations
THE AEROSPACE CORPORATION

Prepared for SPACE AND MISSILE SYSTEMS ORGANIZATION
AIR FORCE SYSTEMS COMMAND
Air Force Unit Post Office
Los Angeles, California 90045

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
U S Department of Commerce
Springfield VA 22151

THE AEROSPACE CORPORATION

Approved for public release; distribution unlimited.

201

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) The Aerospace Corporation San Bernardino, California	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP N/A

3. REPORT TITLE

AEROSPACE SYNTHESIZED AERODYNAMIC HEATING PROGRAM

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

Michael J. Gyetvay and John M. Kohlenberger

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
72 MAR 15	201	16

8a. CONTRACT OR GRANT NO. F04701-71-C-0172	9a. ORIGINATOR'S REPORT NUMBER(S) TR-0172(S2816-75)-1
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) SAMSO-TR-72-181
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Space and Missile Systems Organization Norton Air Force Base, California 92409

13. ABSTRACT

The Aerospace Synthesized Aerodynamic Heating Program is written in the Fortran IV language for the IBM 360-65 computer. This program calculates heating and internal conduction for shapes approximating those of missiles. To do this, it couples various correlations for external high speed convective aerodynamic, for nonaerodynamic, heating with internal transient heat conduction and surface ablation. These may be applied to flat plates, wedges, cones, sphere-cones, spheres, or cylinders with laminar and/or turbulent heat transfer equations applied to the configurations listed. The method is sufficiently general to allow many variations in the physical properties of materials and boundary conditions that one might wish to consider.

Details of the method of analysis are covered in Section 2, while a program listing is in Appendix B, along with sample data cards.

DD FORM 1 NOV 65 1473

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
aerodynamic heating						
high speed convective heating						
surface ablation						
heat conduction						
solar heating						
multi configurations						

UNCLASSIFIED

Security Classification

I-B

Air Force Report No.
SAMSO-TR-72-181

Aerospace Report No.
TR-0172(S2816-75)-1

AEROSPACE SYNTHESIZED AERODYNAMIC HEATING PROGRAM

Prepared by

M. J. Gyetvay and J. M. Kohlenberger

72 MAR 15

San Bernardino Operations
THE AEROSPACE CORPORATION

Prepared for

SPACE AND MISSILE SYSTEMS ORGANIZATION
AIR FORCE SYSTEMS COMMAND
Air Force Unit Post Office
Los Angeles, California 90045

Approved for public release; distribution unlimited.

I-C

FOREWORD

This report by The Aerospace Corporation, San Bernardino Operations, was prepared under Contract No. F04701-71-C-0172 as TR-0172(S2816-75)-1. The Air Force program monitor is Capt. T. W. Swartz. The dates of research for the report include the period July 1963 through April 1971. This report was submitted by the authors April 1971.

The method presented in this report for solving in-depth material heating was first proposed by D. J. Smith and L. Fisher (Ref. 1) of Convair. A preliminary program was formulated at Aerospace by R. Chambers, M. Kausch, and J. Kohlenberger in 1963. The authors have since developed the program into its present form, based on the work of the above named people.

Approved

L. Sashkin

L. Sashkin, Director
Mathematics and Computation Center

E. G. Hertler

E. G. Hertler, Associate Director
Applied Mechanics and Physics Subdivision
Technology Division

Publication of this report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

T. W. Swartz

T. W. Swartz, Capt. USAF (RNSE)

UNCLASSIFIED ABSTRACT

AEROSPACE SYNTHESIZED AERODYNAMIC HEATING PROGRAM, by M. J. Gyetvay and J. M. Kohlenberger

TR-0172(S2816-75)-1
72 MAR 15

The Aerospace Synthesized Aerodynamic Heating Program is written in the Fortran IV language for the IBM 360/65 computer. This program calculates heating and internal conduction for shapes approximating those of missiles. To do this, it couples various correlations for external high speed convective aerodynamic, nonaerodynamic heating, with internal transient heat conduction and surface ablation. The heating may be computed for flat plates, wedges, cones, sphere-cones, spheres, or cylinders with laminar and/or turbulent heat transfer equations applied to the configurations listed. The method is sufficiently general to allow many variations in the physical properties of materials and boundary conditions that one might wish to consider.

Details of the method of analysis are covered in Section 2, while a program listing is in Appendix B, along with sample data cards. (Unclassified Report)

CONTENTS (Continued)

CONTENTS

NOMENCLATURE

A'	=	conductance
Cf	=	friction coefficient
Ch	=	heat transfer coefficient
Cp	=	coefficient of pressure
[illegible]	=	heating multiplication faction
[illegible]	=	enthalpy ratio
[illegible] K	=	conductivity
H	=	heat coefficient based on enthalpy
I	=	enthalpy
J	=	conversion factor
M	=	Mach number
P	=	pressure
Pr	=	Prandtl number
Q	=	heat transfer
$\dot{Q}$	=	heat transfer rate
R	=	radius of nose
Re	=	Reynolds number
St	=	Stanton number
T	=	temperature
U	=	velocity
X'	=	diameter of sphere
Y	=	Griffith-Lewis heating factor as function of Z
Z	=	fictitious wetted distance (see eq. 18)
a	=	nodal thickness
cp	=	specific heat
g	=	gravity (32.2)
r	=	recovery factor
x	=	wetted distance
Δt	=	time step
α	=	angle of attack
ϵ	=	emissivity
θ	=	cone 1/2 angle
Λ	=	sweep angle
μ	=	viscosity
ρ	=	density

SUB AND SUPER SCRIPTS

abl	=	ablation
b	=	base temperature (400 °R)
eff	=	effective
i	=	calculated from reference enthalpy
n	=	a particular nodal point, or time step
r	=	recovery property
s	=	stagnation property
sl	=	sea level
sph	=	sphere property
tan	=	tangency point
v	=	vaporization
w	=	wall property
δ	=	edge of boundary layer
*	=	reference property
∞	=	free stream property

FIGURES

(This page intentionally left blank)

SECTION I

INTRODUCTION

The Aerospace Synthesized Aerodynamic Heating (ASAH) program is written in the FORTRAN IV language for the IBM 360/65 Computer. It is a flexible computer program that couples various correlations for external high speed convective aerodynamic heating with internal transient heat conduction. The flexibility of the program is derived from the many modes of operation available to a user and the ability to run parametric studies in a simple manner. A programmer may select from the following list of body configurations: flat plate, wedge, cone, sphere, cylinder, and nose blunting. An option is available for either ablating or nonablating surfaces. Laminar and/or turbulent heat transfer equations are available for each of the above configurations. Boundary layer transition is based upon either an input transition Reynolds number, an input transition altitude, or in the absence of an input altitude or Reynolds number value, the program will compute its own transition Reynolds number based upon a correlation by McCauley (Ref. 2).

For input, the program requires specification of a trajectory, physical and thermal properties, and control instructions. The trajectory input specifies altitude, velocity and angle of attack as a function of time (Fig. 1-1). Physical dimensions such as distance from forward edge (Fig. 1-2) and material segment thickness, and thermal properties such as material specific heat, density and conductivity must also be supplied. Specific heat, density, and conductivity may be inserted as functions of temperature. Control instructions define modes of operation such as configuration identification, transition criteria, boundary conditions, and computational time interval. The program will compute an optimum integration interval, after the first iteration. All inputs are entered by way of an IBM input routine called NAMELIST, which allows a large degree of freedom as to input format. Heating can be either internally computed aerodynamic, solar, or input as a function of time. Inputs are stored and kept until changed, thus facilitating parametric studies.

Preceding page blank

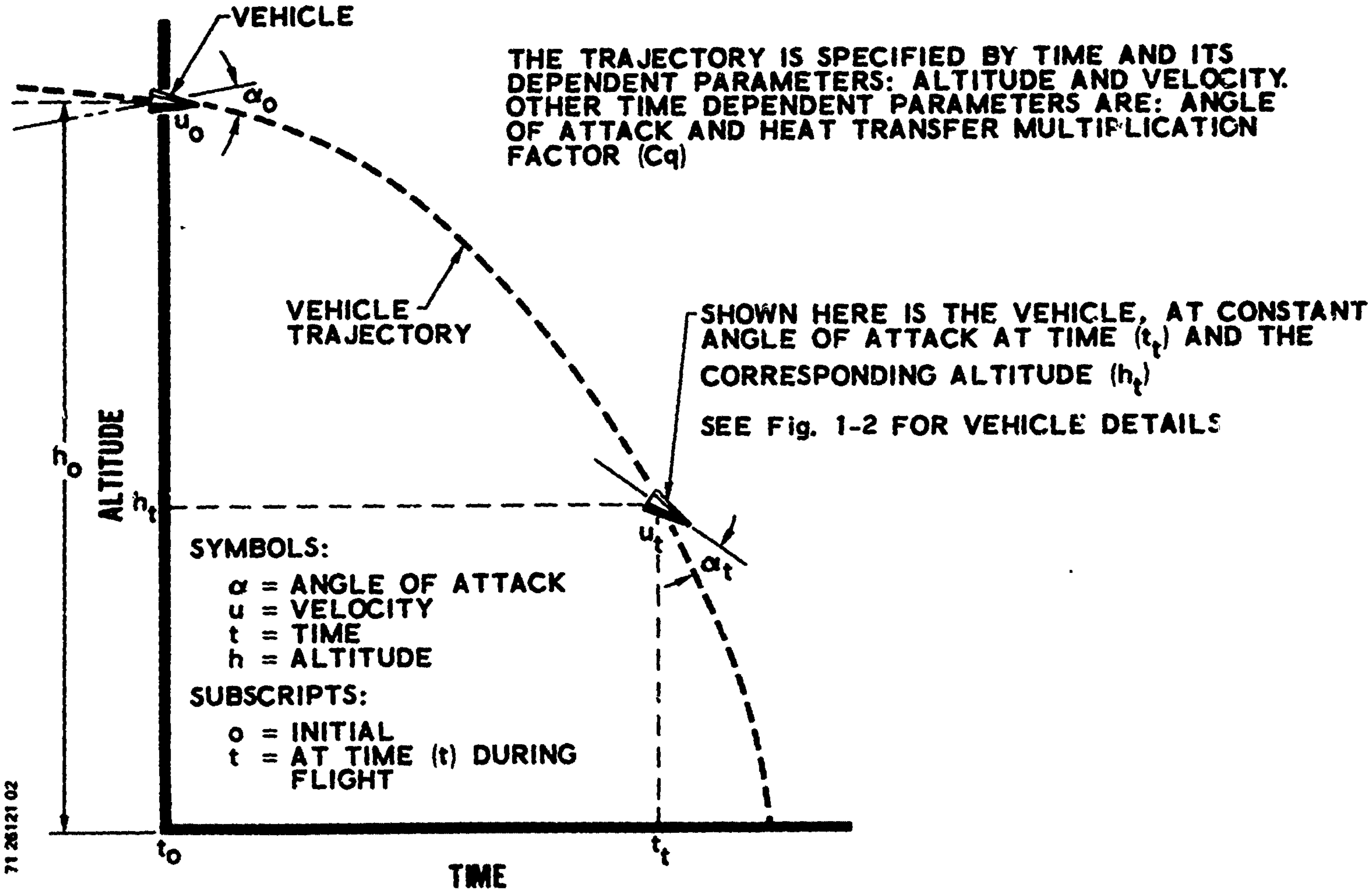


Figure 1-1. Trajectory

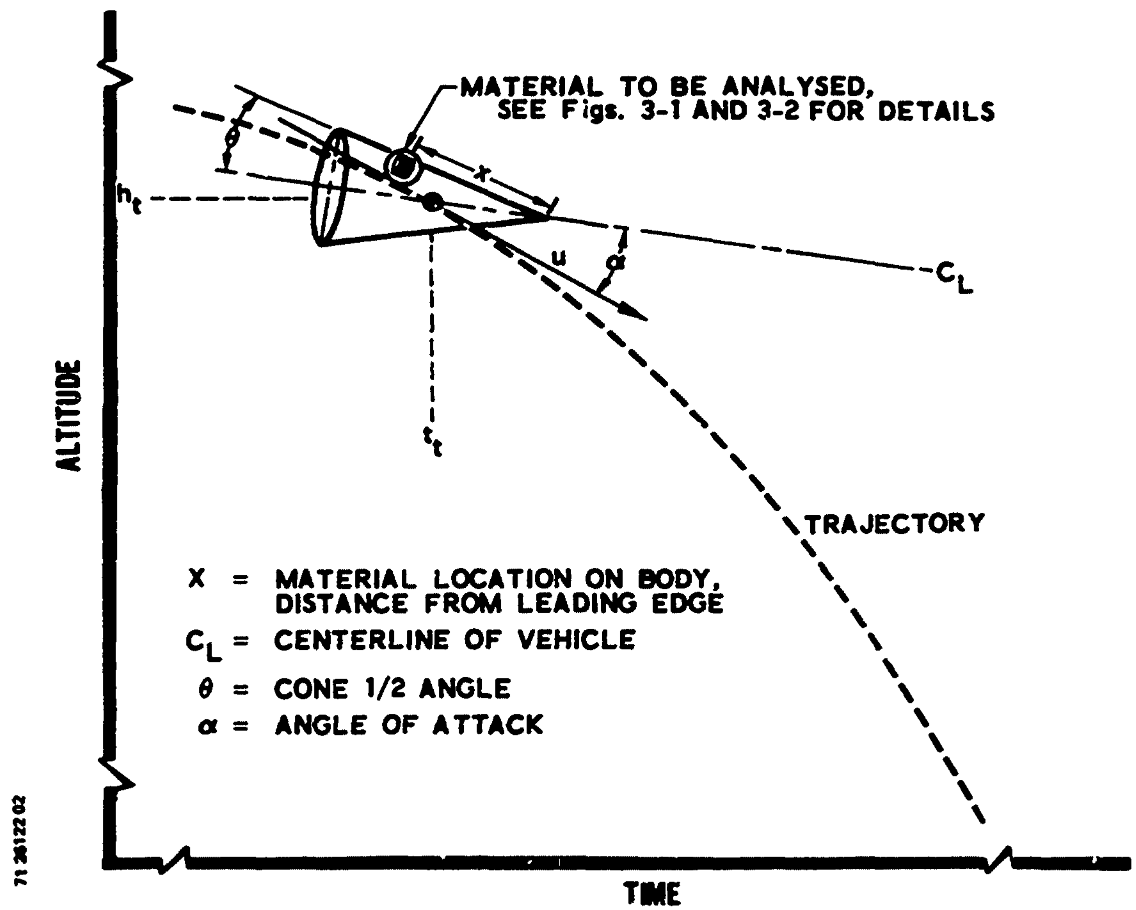


Figure 1-2. Vehicle (1-conical configuration shown)

The printed output for this program includes edge of boundary layer properties, reference conditions, free stream conditions, heat flux, dynamic pressure, Reynolds number, shear, material ablated when applicable, and temperature distribution through the heat protection system. When the nose blunting option is used, radius of a blunt nose, stagnation recession, sonic point heat rate, shear, and heat of ablation for a two-component ablative composite is also provided.

SECTION II

COMPUTATIONAL PROCEDURES

The actual flow of computation through the ASAH program is dependent on the particular problem being solved. A generalized discussion of the flow of typical computations and the subroutines involved will be given in the following paragraphs.

The variable, time, is used as the independent variable to control the overall flow of the program. Subroutine TRAJ linearly interpolates between appropriate trajectory inputs to corresponding free stream velocities and altitudes. These inputs are used in conjunction with subroutine ATMOS to provide the free stream atmospheric properties for any altitude up to 295,278 feet (Ref. 3).

When the wedge or cone shape configurations have been specified, the program algebraically adds the angle of attack to the body half-angle to determine if the SHOCK subroutine or PME (Prandtl-Meyer-Expansion) routine should be used. The SHOCK subroutine uses a similarity parameter to determine the edge of boundary layer stream properties behind the shock. The PME subroutine utilizes the Prandtl-Meyer isentropic relationships to determine fluid properties aft of a sharp corner.

Next, the program computes density, viscosity and Reynolds number. The program uses Eckert's reference enthalpy to evaluate high Mach number heat transfer. These calculations are basically carried out in, or under the control of, subroutine THERMO. The Aerodynamic heating calculation for flat plates, wedges and cones are carried out in subroutine ECKERT. Subroutine ECKERT uses the modified Reynolds analogy to correlate skin friction with heat transfer. In the case of laminar flows, skin friction is evaluated by the Blasius relationship. When the flow is turbulent, skin friction is determined using the Schultz-Grunow relationship. The aerodynamic heating calculations for spheres, cylinders, and blunted cones are handled in subroutine KEMP, using the Kemp-Riddell stagnation correlation (Ref. 4), or the Sibulkin equation for Mach numbers less than 3. There are also options for calculating heating inputs using the Hankey sonic point heating formulation, in subroutine HANKEY. Another option is available in subroutine CONSTQ. This routine provides heating values based on interpolation from tabular input.

Upon completion of the external heating calculations, the program calls on subroutine ENERGY. This routine performs an energy balance computation at the flow field - material interface.

The thermal protection system (heat sink or ablating surface) is divided into a number of segments. With the net heat load as a boundary condition, Fourier's transient conduction heat transfer equation is solved in finite-difference form. The calculations are carried out under the control of subroutines ONED or TWOD for either one or two spatial dimensions.

The one-dimensional option allows the heat protection system to be divided into a maximum of 40 segments. The two-dimensional option allows the system to be characterized by a 20 x 20 matrix of points. Material specific heat and conductivity can be inserted as functions of temperature. Each matrix element or segment may have different material properties. The program obtains particular values of the material properties, for use in computation, by accessing subroutine GETCPK or GPCPK2. These routines access the proper input material properties tables as a function of temperature and provide the needed values.

When the ablation option is exercised, the energy balance at the body surface allows for energy to be absorbed by the ablating material. This adjustment is carried out in subroutine ABLATE. A gross effective heat of ablation is computed as a function of heat of vaporization and enthalpy difference. The rate at which material is removed from the surface is determined by the difference between net heat flux to the surface and the heat conducted away into the next material segment. When the surface temperature reaches the ablation temperature and net heat flux to the surface is positive, the ablation is performed.

The nose-blunting option uses the Kemp stagnation point heating and engineering correlations to obtain heating at the sonic or tangency positions. These calculations are carried out under the control of subroutine ABLUNT.

The computations that have been, so far, described comprise those required to advance one step in time. The main program now performs various checks relating to the print and punch options. It then does a stability check to select a proper

time step for the next step in the computation. Control is then transferred to the beginning of the computational loop and reenters subroutine TRAJ. This mode of operation is continued until such time as the case is completed, or an unrecoverable problem is encountered. When this occurs, the program transfers to a call to subroutine INPUT to look for subsequent cases.

(This page intentionally left blank)

SECTION III

METHOD OF ANALYSIS

This section of the report reviews the various methodologies that are used by the computer program to handle the individual aspects of the overall problem. The intent is to document the equations and data that are in the program and indicate the source of major portions of the formulation. No attempt is made to derive the equations used or to justify the assumptions inherent in the computer program.

3.1 HEATING RATES FOR FLAT PLATES, WEDGES AND CONES

The reference enthalpy method proposed by Eckert (Ref. 5) is used to compute the convective aerodynamic heating rates. This method is based on the assumption that the incompressible mass, momentum, and energy equations can be used for compressible flow solutions, providing the transport properties of the gas are known and evaluated at a reference enthalpy. Real gas effects, including gas disassociation and ionization, must be taken into account also in the determination of the thermodynamic properties of the gas just outside the boundary layer.

3.1.1 Laminar Boundary Layer

The Blasius solution for an incompressible boundary layer is

$$St = \frac{Cf}{2} Pr^{-2/3} \tag{1}$$

and

$$Cf = \frac{0.664}{\sqrt{Re}} \tag{2}$$

where St is the Stanton number, while Pr is the Prandtl number (= 0.65) and $Re = \rho UX'/\mu$ is the Reynolds number. Using Eckert's reference enthalpy

$$Re^* = \frac{\rho^* UX'}{\mu^*} \tag{2a}$$

$$H_i = 3600 \; St \, \rho_\delta U_\delta \, g \tag{3}$$

Preceding page blank

where ρ and μ are evaluated at a reference enthalpy, I^*, and U is evaluated just outside the boundary layer, and g is gravity.

When computing recovery enthalpy the recovery factor is Pr^n with $n = 1/2$ for laminar flow and $n = 1/3$ for turbulent flow. Since Pr was assumed = 0.65 the following values for recovery factor (r) are used:

= 0.866 for turbulent flow

= 0.806 for laminar flow in nonstagnation areas

= 1.0 in stagnation areas (sphere and cylinder).

$$\dot{Q} = H_i (I_r - I_W) \tag{4}$$

The thermodynamic properties of the air ahead of the shock, P_∞, T_∞, and the free stream velocity and Mach number, U_∞ and M_∞, can be determined from the given flight conditions. An approximation of the ARDC 1959 atmosphere is used. The free stream and wall enthalpies can be determined once the ambient temperature and wall surface temperature are known. The thermodynamic properties just outside the boundary layer are determined from the properties given by Schadt and Livett (Ref. 6) once the hypersonic similarity parameters, $M_\infty \sin\alpha$ and $M_r \sin\alpha$ are known (α, here is total of $\alpha + \theta$). Eckert states that

$$I^* = I_\delta + 0.5\,(I_W - I_\delta) + 0.22\,(I_r - I_\delta) \tag{5}$$

where

$$I_r = I_\delta + \frac{rU_\delta^2}{2Jg} \tag{6}$$

The transport properties are given by Hansen (Ref. 7). Curve fit equations were written making ρ^* and μ^* dependent on temperature. The terms ρ_b and μ_b are evaluated at a base temperature equal to 400°R suggested by M. F. Romig (Ref. 8). For cones the following correlations are used to define edge flow field properties:

when $M_r \sin\alpha < 4.0$:

$$(U_\infty^2 - U_\delta^2) \times 10^{-6} = -0.107\,(M_r \sin\alpha)^3 + 1.91\,(M_r \sin\alpha)^2 + 0.04\,M_r \sin\alpha + 1.57 \tag{7}$$

when $M_r \sin\alpha \geq 4.0$:

$$(U_\infty^2 - U_\delta^2) \times 10^{-6} = -0.0062\,(M_r \sin\alpha)^3 + 1.414\,(M_r \sin\alpha)^2 + 1.038\,M_r \sin\alpha + 6.187 \qquad (8)$$

with $M_r = U_\infty/1086$.

The following hypersonic similarity equations for $\frac{P_\delta}{P_\infty}$ and $\frac{I_\delta}{I_\infty}$ are then employed:

when $1.5 \leq M_\infty \sin\alpha \leq 5.0$

$$\frac{P_\delta}{P_\infty} = 0.0489\,(M_\infty \sin\alpha)^3 + 1.06\,(M_\infty \sin\alpha)^2 + 1.16\,(M_\infty \sin\alpha) \qquad (9a)$$

when $M_\infty \sin\alpha > 5.0$

$$\frac{P_\delta}{P_\infty} = 0.0216\,(M_\infty \sin\alpha)^3 + 0.8373\,(M_\infty \sin\alpha)^2 + 4.177\,(M_\infty \sin\alpha) - 3.182 \qquad (9b)$$

and when $M_\infty \sin\alpha \leq 8$.

$$I_\delta = I_\infty\,[1.03 + 0.0827\,M_\infty \sin\alpha + 0.2174\,(M_\infty \sin\alpha)^2 - 0.0006956\,(M_\infty \sin\alpha)^3] \qquad (10a)$$

or, when $M_\infty \sin\alpha > 8$.

$$I_\delta = I_\infty\,[1.106 - 0.3685\,M_\infty \sin\alpha + 0.3466\,(M_\infty \sin\alpha)^2 - 0.007766\,(M_\infty \sin\alpha)^3] \qquad (10b)$$

with $M_\infty = U_\infty/1086$.

Similar correlations are used to compute edge of boundary layer fluid properties for a wedge geometry. In the case of the flat plate P_∞ and U_∞ are used in place of P_δ and U_δ; the ambient air properties are assumed at the edge of the boundary layer. The density is obtained from the ideal gas equation, and the viscosity from the Sutherland equation. The heat transfer parameter obtained by Eq. (3) is multiplied by a factor of 1.73 for conical flow with laminar conditions (Mangler transformation).

3.1.2 Turbulent Boundary Layer

The Schultz-Grunow correlation for an incompressible turbulent boundary layer is used with the modified Reynolds analogy to predict turbulent heat rate at high velocities

$$St = \frac{Cf}{2} \, Pr^{-2/3} \tag{11}$$

with

$$Cf = \frac{0.370}{[\log(Re^*)/2.303]^{2.584}} \tag{12}$$

The equation for the heat transfer parameter can be reduced to the form

$$H_i = 3600 \, g \, St \, U_\delta \, \rho_\delta \tag{13}$$

and

$$\dot{Q} = H_i \, (I_r - I_w) \tag{14}$$

$$Re^* = \frac{\rho^* \, U_\delta \, X'}{\mu^*} \tag{15}$$

For a cone the heat transfer parameter is increased by a factor of 1.176 (Mangler transformation) over the flat plate heat transfer rate.

Hypersonic similarity parameter correlations yield I_δ, U_δ, P_δ. Other fluid properties at the edge of the boundary layer, are evaluated with the following real gas correlations:

when $I_\delta < 1300$

$$\rho_\delta = 5.76 \times 10^{-5} \, I_\delta^{-0.849} \, P_\delta \tag{16a}$$

$$\mu_\delta = 4.2642 \times 10^{-8} \, I_\delta^{0.493} \tag{16b}$$

when $I_\delta \geq 1300$

$$\rho_\delta = 8.65 \times 10^{-6} \, I_\delta^{-0.584} \, P_\delta \tag{17a}$$

$$\mu_\delta = 2.8428 \times 10^{-7} \, I_\delta^{\,0.228} \tag{17b}$$

$$Re_\delta = U_\delta \rho_\delta X' / \mu_\delta \tag{18}$$

This local Reynolds number is tested against a transition criteria to determine laminar or turbulent flow.

For laminar and turbulent flow, X' the input wetted-distance, is used to compute local Reynolds number.

3.3 SPHERE-CONE

3.2.1 Laminar Flow

When the flow over a sphere-cone is laminar, the heating correlation is identically that recommended by Griffith-Lewis (Ref. 9).

Local flow properties are evaluated at a nondimensionalized length:

$$Z = \frac{\theta^2 \cos\theta \, X}{0.802 \, R} \tag{19}$$

The following array of algebraic equations are solved to determine a local heating rate. The factor Y is determined from table lookup and is identical to the valve, suggested by reference 9.

$$I_\infty = 0.24 \, T_\infty \tag{20}$$

$$I^* = \frac{I_\delta}{6} \left(1 + 3 \frac{I_w}{I_\delta} \right) \tag{21}$$

$$Cs2 = \frac{2.58 \, I_\infty^{\,0.253}}{I^{*\,0.538}} \tag{22}$$

$$\mu_\infty = 0.173 \times 10^{-5} + 0.00833 \times 10^{-5} \, I_\infty \tag{23}$$

$$Re_\infty = \frac{64.4 \, \rho_\infty U_\infty R}{U_\infty} \tag{24}$$

Then, if $Z \leq 1.5$ the coefficient of pressure is defined as

$$Cp = 2.46\theta^2 \tag{25a}$$

or, if $Z > 1.5$

$$Cp = 2\theta^2 (-6.3Z^5 + 29.3Z^4 - 52.2\,Z^3 + 43.15Z^2 - 15.5Z + 2.76) \tag{25b}$$

$$Ch = \frac{M_\infty Y\theta^2 \, Cs2 \left(1.0 + \dfrac{1.428}{M_\infty^2 Cp}\right)}{0.632\sqrt{Re_\infty}} \tag{26}$$

Surface heating is then computed as

$$H_i = \frac{1.2288 \times 10^6 \, ChP_\infty M_\infty^2}{U_\infty} \tag{27}$$

Finally, the convective heat transfer, to a surface of unit area, is computed by the equation

$$\dot{Q} = H_i (I_r - I_w) \tag{28}$$

3.2.2 Turbulent Flow

For turbulent flow on a sphere-cone, a Schultz-Grunow heating solution is employed. The flow field properties are computed by assuming isentropic expansion from the stagnation point to a given pressure. This pressure is obtained from the correlation suggested by Griffith and Lewis. The local flow field properties are computed as follows:

$$P_\delta = P_\infty (0.7\,M_\infty^2 \, Cp + 1.0) \tag{29}$$

$$U_\delta = \sqrt{\frac{0.8333\,U_\infty^2 \left(1.0 - \dfrac{P_\delta}{P_\infty}\right)^{0.286} \left(1.168\,M_\infty^2 - 0.1668\right)^{0.714}}{M_\infty^2}} \tag{30}$$

$$I_s = I_\infty + \frac{U_\infty^2}{50061.5} \tag{31}$$

$$I_\delta = I_s - \frac{U_\delta^2}{50061.5} \tag{32}$$

at this point Eckert's reference enthalpy, Eqs. (5) and (6) are used along with Eqs. (11) through (17).

The turbulent heat transfer coefficient to a cone is determined by increasing the flat plate correlations by $(2)^{1/5}$. Finally the convective heat transfer to a surface of unit area is computed from Eq. (27).

3.3 SPHERE AND CYLINDER

The heat transfer to the stagnation point of a sphere given by the Kemp and Riddell equation is used:

$$Q_{sph} = \frac{89.64 \times 10^6}{\sqrt{X'}} \sqrt{\frac{\rho_\infty}{\rho_{sl}}} \left(\frac{U_\infty}{26000}\right)^{3.15} \left[\frac{(i_s - i_w)}{(i_s - 129.5)}\right] \tag{33}$$

where ρ_{sl} is sea level density.

An approximation to the Sibulkin equation (Ref. 10) is used for $M_\infty \leq 3.0$

$$Q_{sph} = 174.0\sqrt{\frac{\rho_\infty U_\infty}{X'}} \; (i_s - i_w) \tag{34}$$

On an unswept cylinder, the convective heat transfer to the stagnation point is

$$Q_{cyl} = Q_{sph} \times \text{factors} \tag{35}$$

The following equations are used to account for sweep angle effects

$$Q_{cyl} = 0.931 \; Q_{sph} \, (\cos\Lambda_{eff})^{1.5} \tag{36}$$

where

$$\Lambda_{eff} = \mathrm{Tan}^{-1}\left[\frac{\cos\alpha \sin\Lambda}{\sqrt{1.0 - (\cos\alpha \sin\Lambda)^2}}\right] \tag{37}$$

A program option is available whereby Q_i, a heating input, can be found by way of a table look-up. If Cq is zero or negative, the program looks up the proper Q_i and sets Cq = 1 at which point QNET is found by the ordinary external energy balance used for all situations.

$$\mathrm{QNET} = (\mathrm{Cq}\, Q_i) - \left[0.174\epsilon\left(\frac{T_w}{100}\right)^4\right] \tag{38}$$

3.4 TEMPERATURE DETERMINATION

The material temperature is a function of both the net heat transfer to the wall (i.e., including radiation losses) and the diffusivity properties of the material. In order to evaluate the surface temperature and the material temperature gradients, the so called "lumped parameter" method is employed. The following section is a general description of this method.

The material is divided up into an arbitrary number of small segments, both perpendicular and parallel to the flow direction as shown in Figs. 3-1 and 3-2. Then for some small time increment (controlled by stability considerations) the net heat flux to the surface is determined. Knowing this value, the net heat transfer from segment (1) to all the adjacent segments is computed for the same small time increment. In this calculation it is assumed that all the mass is concentrated in a point source at the center of the segment and that the heat transfer takes place over an area equal to the interface between the segments and a distance equal to the spacing between the centers of the segments. The difference between the heat transferred to segment (1) by convection and radiation, and that which is transferred to all the adjacent segments by conduction causes a certain temperature rise in segment (1) during the calculation time interval. By making the calculation time interval and the size of the segments very small, the accuracy of the method may be made as good as is desired within the accuracy of the computer being used.

A forward marching explicit method of computing the temperatures is used where the temperature of a given segment at the end of a time interval is dependent upon the temperature of the given segment and adjoining segments at the beginning of the time interval.

3.4.1 One-Dimensional Conduction

The material or materials may be divided into n segments, as shown in Fig. 3-1, where $n \leq 40$. They are numbered from the outer surface inward.

The heat balance for conduction may be written as follows:

for internal segments,

$$T'_n = T_n + A' \frac{T_{n-1} - T_n}{\frac{1}{2}\left(\frac{a_n}{K_n} + \frac{a_{n-1}}{K_{n-1}}\right)} + \frac{T_{n+1} - T_n}{\frac{1}{2}\left(\frac{a_n}{K_n} + \frac{a_{n+1}}{K_{n+1}}\right)} \tag{39}$$

for the surface nodal point (n = 1)

$$T'_n = T_n + A' \left\{ \frac{Q_{net}}{12} + \frac{T_{n+1} - T_n}{\frac{1}{2}\left(\frac{a_n}{K_n} + \frac{a_{n+1}}{K_{n+1}}\right)} \right\} \tag{40}$$

Where T'_n is the temperature at segment n for the succeeding time increment, "a" is the segment thickness, and K is conductivity. The conductance is defined as:

$$A' = \frac{\Delta t}{25.0\ \rho_n\ cp_n\ a_n} \quad . \tag{41}$$

For the last segment a quantity HINSD = 12 (TINSD - T_n) is used to account for internal heating (if present).

3.4.2 Two-Dimensional Conduction

The two-dimensional heat conduction technique involves setting up a structure in a matrix such that the maximum number of rows = I and maximum number of columns = K. The material must be homogeneous in the X direction but not necessarily in the Y direction. This is shown in Fig. 3-2.

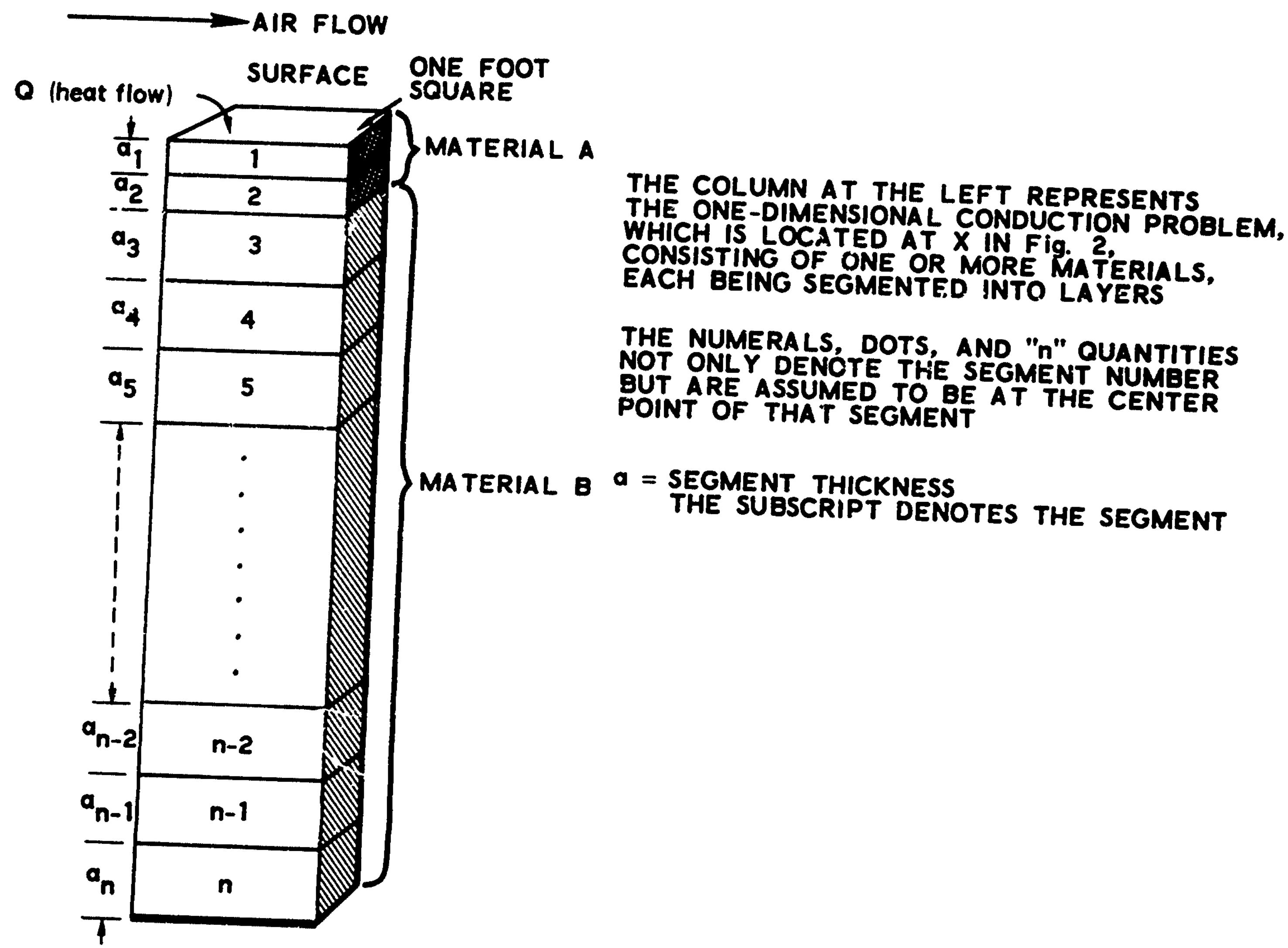


Figure 3-1. One-Dimensional Material

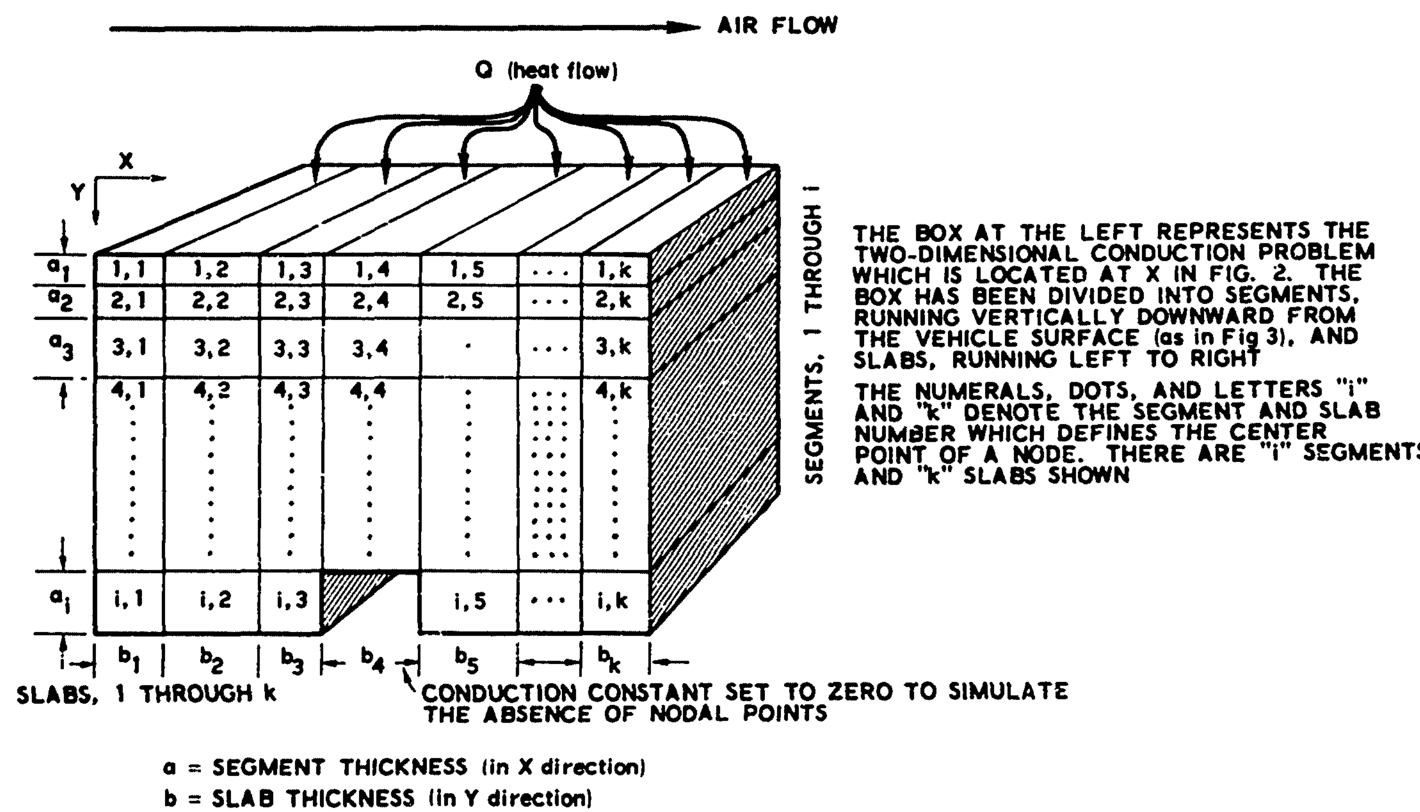


Figure 3-2. Two-Dimensional Material

The first slab in successive rows is numbered (1, 1), (2, 1),.. (I, 1), there are IxK nodes, or elements, therefore the temperature at a node i, k, time $t + \Delta t$, is dependent upon the temperatures of the contiguous segments at time t. The general picture for a given nodal point is shown in Fig. 3-3.

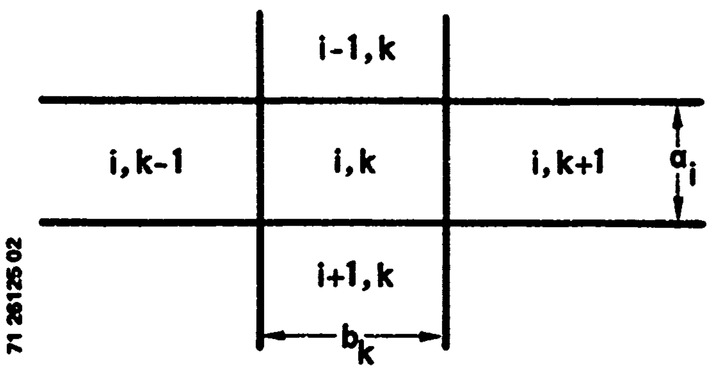


Figure 3-3. Nodal Subscript Relationships

As shown in Fig. 3-3, the equations for finding the temperature $T'_{n_{i,k}}$ for segment (i, k) are:

$$\text{FACTOR} = \frac{\Delta t}{25\,\rho_i\,cp_{i,k}\,a_i\,b_k} \tag{42}$$

$$\text{TERM 1} = \frac{b_k\,CC_{i-1,k}}{\frac{1}{2}\left(\frac{a_{i-1}}{K_{i-1,k}} + \frac{a_i}{K_{i-1,k}}\right)}\,(T_{i-1,k} - T_{i,k}) \tag{43}$$

where K is conductivity, and CC is a multiplying constant that allows configuration flexibility,

(CC = 1 if conduction desired and 0 if no conduction),

and $(T_{i-1,k} - T_{i,k})$ = heat from (i-1, k) segment (44)

TERM 2 in like manner with heat from (i+1, k) segment

TERM 3 in like manner with heat from (i, k-1) segment

TERM 4 in like manner with heat from (1, k+1) segment

$$T'_{n_{i,k}} = FACTOR(TERM1 + TERM2 + TERM3 + TERM4) \tag{44}$$

3.5 ABLATION TECHNIQUES

Some of the more promising types of heatshield materials for high speed flight include those whose surfaces melt or ablate because of the aerodynamic heating. In this kind of process it is necessary to compute the rate of surface ablation. This is done in the following manner. The heat transfer to the ablating surface is calculated as previously until the surface segment reaches ablation temperature. At this time two simplifying assumptions are invoked:

(1) The segment temperature is maintained constant and equal to the ablating temperature. This is usually considered to be the melting temperature of the fibers (glass, quartz, asbestos, etc.) rather than the melting temperature of the binder.

(2) The integrated heat flux to the segment is compared against the effective heat of ablation of the material. The effective heat of ablation is determined from test data and numerically represented by the following equation:

$$H_{abl} = H_v + \eta \, (I_r - I_w) \tag{45}$$

H_v and η are input parameters to the code.

Material weight loss is determined by integrating in time. When weight loss equals equivalent weight of a segment thickness that nodal point is removed from the solution temperature matrix.

3.6 NOSE BLUNTING TECHNIQUE

Nose blunting is accomplished with a two point computation. The assumption is made that a hemispherical geometry is always maintained during ablation, and that the ablating material is uniform throughout. The heat entering the stagnation point is found by means of the Kemp stagnation heating Eq. (32).

When the temperature of the outermost segment reaches ablation temperature, a new radius is calculated by the following equations using an effective heat of ablation to model surface recession. The effective heat of ablation formulation allows for a reduction of that parameter as a function of pressure or shear.

3.6.1 Laminar Flow

For laminar flow, the heating to the tangent point is computed with the correlation suggested by Stetson (Ref. 12). After determining the heat rate ($\dot{Q}_{tan}$), from a Ref. 11 table look-up, and $H_{abl_{min}}$ which is the minimum heat of ablation due to either shear or pressure degradation at both the stagnation and tangency point, the following equations are used:

$$DF = \frac{Q_{sph}\,\Delta t}{\sqrt{R_n}\,\rho\,(H_{abl})_{stag}} \quad \text{for recession, and for nose radius change} \tag{46}$$

$$R_n = R_{n-1} + DF\left[\frac{\sin(\theta)}{1.0 - \sin(\theta)} - \frac{Q_{TAN}\left(H_{abl}\right)_{stag}}{1.0 - \sin(\theta)\left(H_{abl}\right)_{tan}}\right] \tag{47}$$

Or, in the case where Stetson calculations are not prescribed:

$$FDF = 0.637\ DF \tag{48}$$

and Eqs. (51) and (52) are used.

3.6.2 Turbulent Flow

For turbulent flow, the heating at the sonic point is computed using a correlation suggested by Hankey (Ref. 11), which is:

$$Q_i = \frac{18000\left(\frac{\rho_\infty}{\rho_{sl}}\right)^{0.8}\left(\frac{U_\infty}{1000}\right)^{3}\left(1 - \frac{i_w}{i_s}\right)}{R^{0.2}} \tag{49}$$

Equations (33) and (46) are again employed, with the following modifications to their results to reflect geometric changes.

$$DFF = 1.0 - DF/R \tag{50}$$

where

$$DF = \frac{17600\sqrt{\frac{\rho_\infty}{\rho_{sl}}}\left(\frac{U_\infty}{26000}\right)^{3.15}\left(\frac{I^* - I_w}{I^* - 129.5}\right)\Delta t}{\sqrt{R}\ \rho_i\ H_{ABL}} \tag{50a}$$

$$FDF = 1.0 - \frac{DF\left[4.68\left(\frac{\rho_\infty}{\rho_{sl}}\right)^{0.3}(R)^{0.3}\right]}{R} \tag{51}$$

$$FD = \left(1.0 - \frac{FDF}{R}\right)0.74 \tag{51a}$$

Then the new nose radius is computed:

$$R_n = R_{n-1}\ \frac{FDF^2 - 2\ DFF\ FD + DFF^2}{2.0\ (DFF - FD)} \tag{52}$$

SECTION IV

SUBROUTINE DESCRIPTION

4.1 INPUT

This is the routine for listing the data cards and then reading them in according to NAMELIST "&BD06ØS." Input also makes minor adjustments on those data, such as filling in the QCØNS table with 1.0 when left blank.

4.2 OUTINT

This routine prints out the integer values, which were read in, before the computations start to allow easy inspection of that data input.

4.3 OUTPUT

This routine, following OUTINT, prints the rest of the data, as input.

4.4 TRAJ

Linear interpolation for tabular input data. The program increments time which is the controlling element in the tabular-interpolations. The tabular data evaluated within this routine include: altitude, velocity, angle of attack, internal convective coefficient, inside temperature and QCØNS.

4.5 ATMOS

The atmosphere subroutine determines the free stream pressure, density and temperature. The 1959 ARDC atmosphere (Ref. 13) has been assumed in this routine. The internal programming uses altitude in meters for the independent parameter table. Pressure and temperature are determined by exponential and linear curve fits valid over a limited altitude range. The coefficients to those curve fits are reevaluated at 33.5, 76.2, 143.5, 161.5, 241 and 275 Kft. Free stream density is computed using the perfect gas relationship. Internally the units are slugs/ft^3. Figures 4-1 and 4-2 show the free stream pressure and temperature as a function of altitude.

Preceding page blank

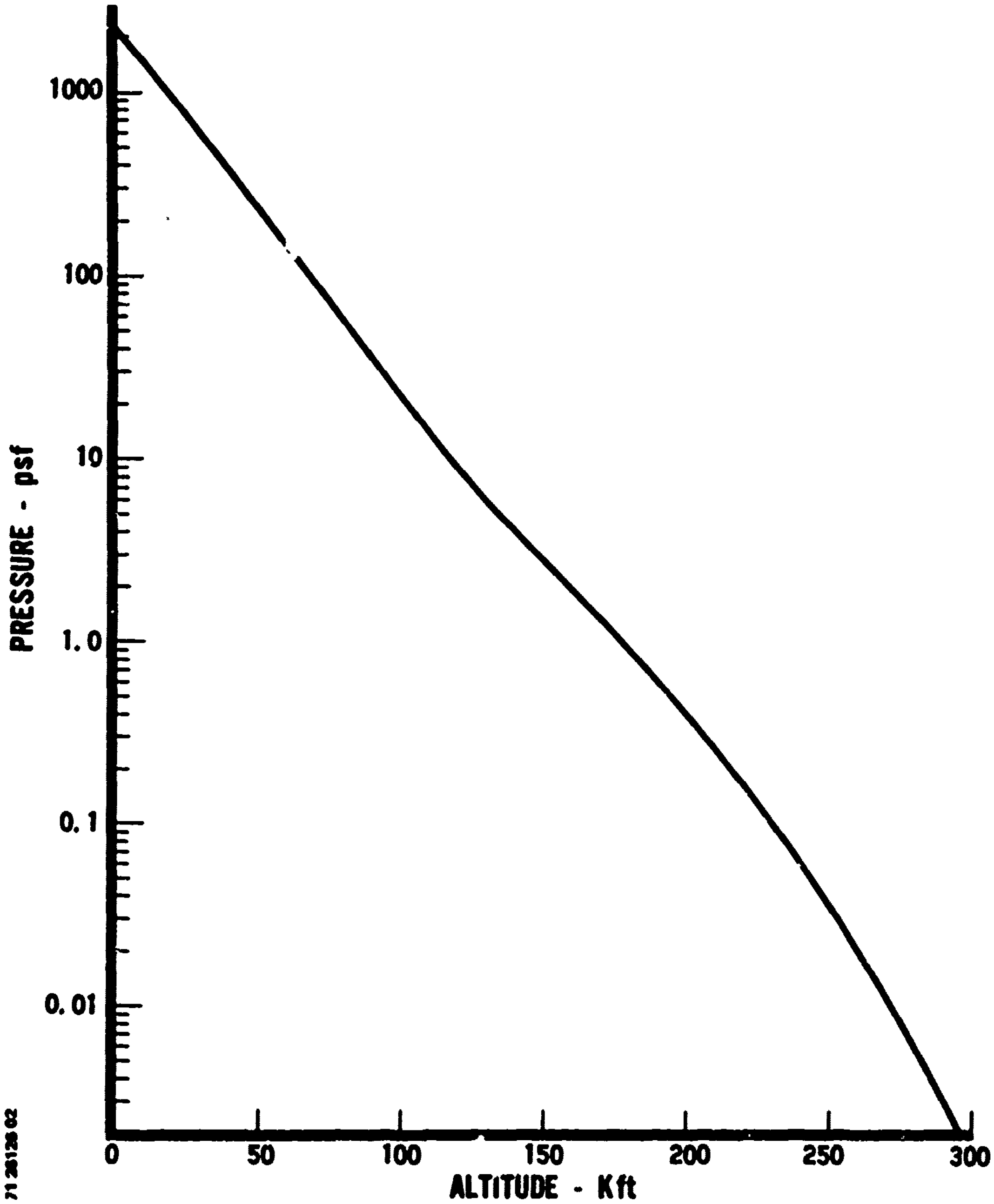


Figure 4-1. Pressure vs Altitude - 1959 ARDC Atmosphere

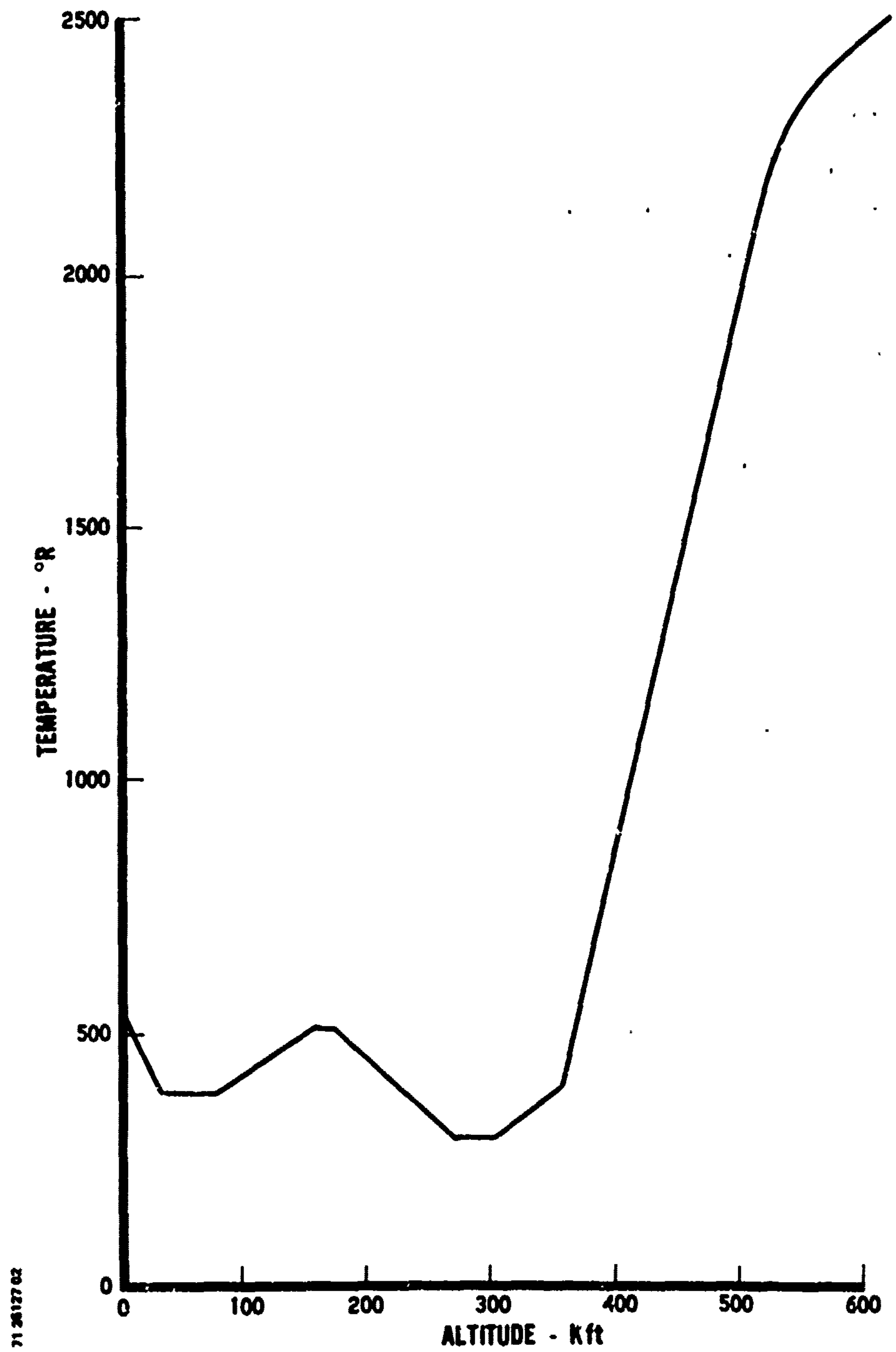


Figure 4-2. Temperature vs Altitude - 1959 ARDC Atmosphere

4.6 PRANDTL-MEYER EXPANSION (PME)

To enter the PRANDTL-MEYER EXPANSION routine the algebraic sum of the angle of attack and cone half-angle must be negative and the configuration must be either flat plate, cone, or wedge. Therefore, to enter this routine, angle of attack (AOFA) must be input as a negative term with either the flat plate, wedge or cone options. The PRANDTL-MEYER equations used in this subroutine assume a $\gamma = 1.4$. The equation used to determine the initial angle associated with the given free stream Mach number is:

$$\mathrm{UNI} = \nu = 140.3 \tan^{-1}(0.408\beta) - 57.3 \tan^{-1}\beta \tag{53}$$

where

$$\beta = \left[M_{\infty}^{2} - 1\right]^{1/2} \tag{54}$$

The expansion angle of interest is $\mathrm{UNE} = \mathrm{UNI} - \lambda + \alpha$ (55)

Three curve fits were used to determine the expansion Mach number:

1. For UNE $\leq 50°$

$$\mathrm{AME} = \frac{-26.3}{\mathrm{UNE} - 81.77} + 0.7895 + 0.0279 + \mathrm{UNE} \tag{56}$$

2. For $50° <$ UNE $\leq 102.32°$

$$\mathrm{AME} = \frac{-280.09}{\mathrm{UNE} - 130.20} + 0.8941 + 8.28 \times 10^{-3}\,\mathrm{UNE} \tag{57}$$

3. For UNE $> 102.32°$

$$\mathrm{AME} = -\frac{284.85}{\mathrm{UNE} - 130.43} - 1.0199 + 8.68 \times 10^{-3}\,\mathrm{UNE} \tag{58}$$

Herein remains a problem. The free-stream Mach numbers are usually high, hence the third equation is usually used and, in fact, its limit of applicability is exceeded if unbounded. A SEPARATION stop should be placed upon this equation and a ratio limit of expansion pressure to upstream pressure imposed. Internally this limit is identified as "CONPR."

Once the pressure ratio is determined, pressure, temperature, density and velocity are determined using ideal gas isentropic expansion relationships with Mach number as the independent parameter. Reference viscosity and reference density, needed for the heating calculations, are computed from enthalpy and pressure dependent curve fits identical to those used in the shock subroutine.

4.7 SHOCK

The control subroutine calls the SHOCK subroutine if the configuration input is either flat plate, wedge, or cone, and if the algebraic sum of the cone half-angle and angle of attack is positive. In this routine the local (edge of boundary layer-inviscid, noninteraction) flow field properties, velocity, pressure and temperature, are computed. For a flat plate at zero angle of attack or for a wedge and cone at an effective angle of zero degrees to the stream flow, the local flow field properties are set equal to the free stream properties. If the configuration has any net positive angle of incidence to the free stream, oblique shock flow field data are computed for the effective body streamline. The basic property data utilized in this portion of the routine were based upon the real gas work of M. Romig. (Ref. 15) The equations are power series using the hypersonic similarity parameter as the independent parameter. The correlations were performed by Schadt and Livett of General Dynamics. (Ref. 13) The hypersonic similarity parameter for the local pressure is the product of free stream Mach number and sine of the cone half-angle. The parameter for velocity is the product of reference Mach number and sine of the cone half-angle. Figures 4-3 and 4-4 present pressure ratio and velocity difference across a shock developed from a wedge configuration. Figures 4-6 and 4-7 show similar results for the cone configuration. The correlation equations actually used within the program, and their range of applicability, are illustrated on each figure. Note that the local property data presented by these figures yield values at the edge of the boundary layer and not immediately behind the shock. Observe also that these data are for sharp bodies only or at that axial station where sharp cone conditions exist. This distance can be established from the entropy swallowing work done by Jacobs (Ref. 14).

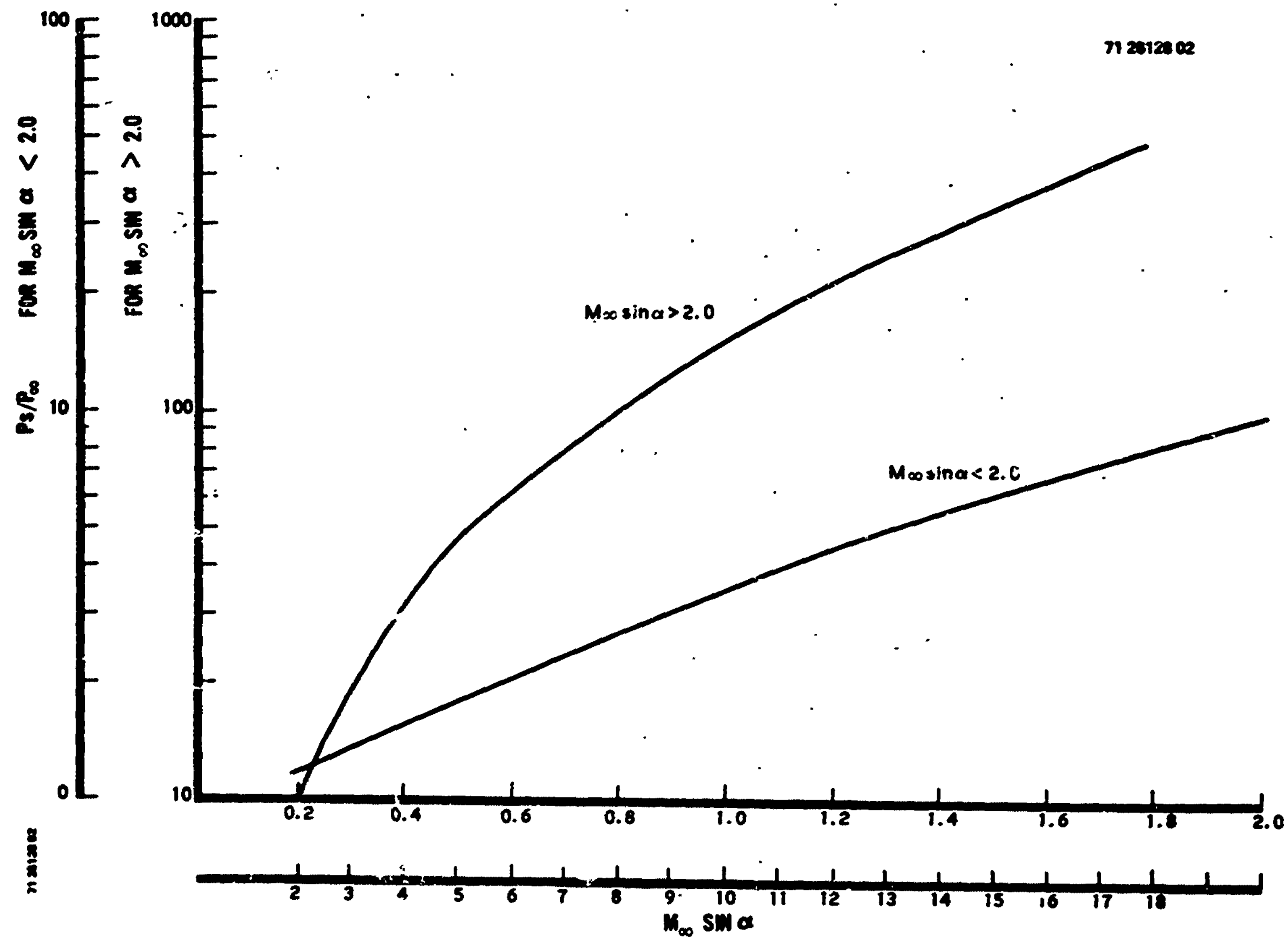


Figure 4-3. Wedge Pressure Ratio

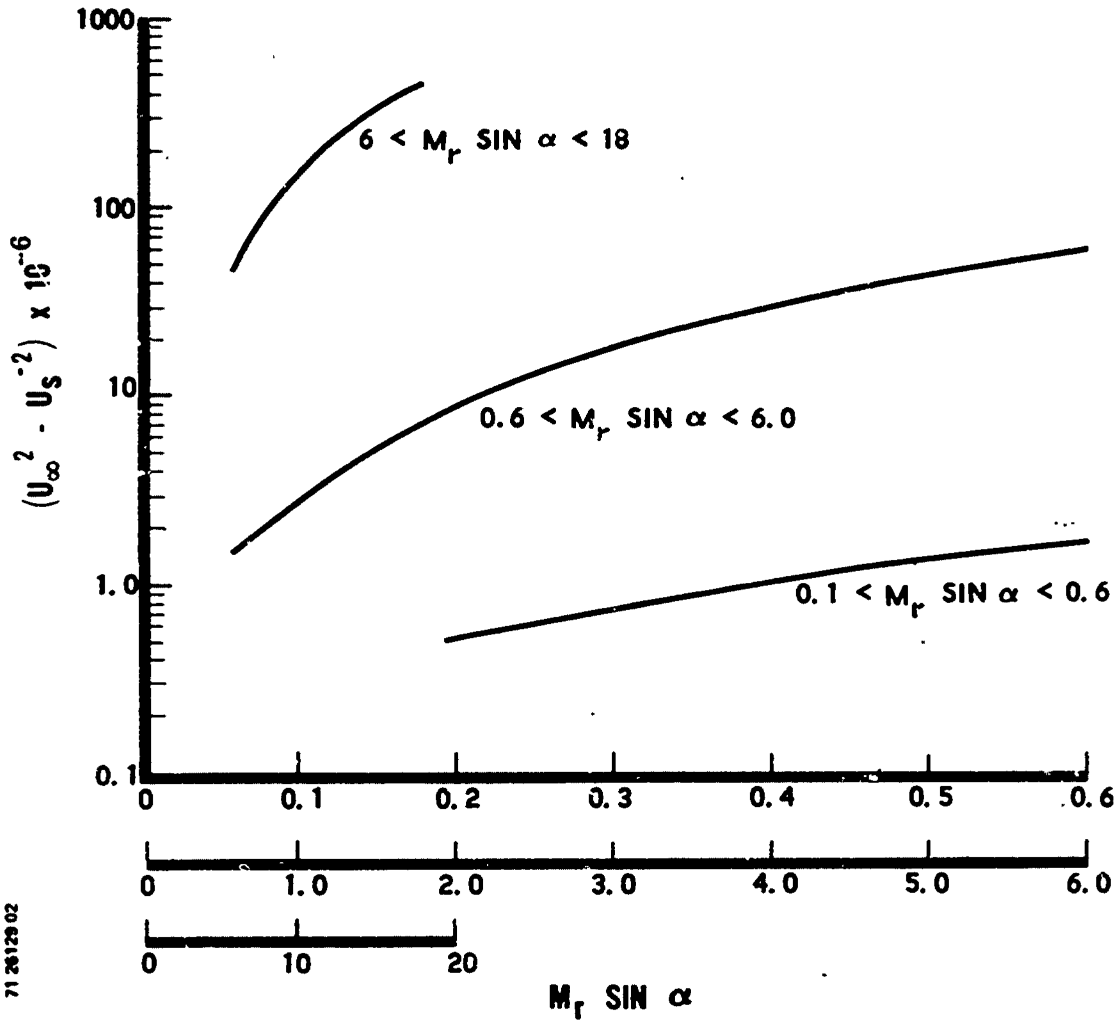


Figure 4-4. Wedge Velocity Parameter

4.8 THERMO

The THERMO subroutine computes enthalpy for the sharp cone and sharp wedge configurations. The enthalpy change across the shock is determined from correlations of M. Romig's work (Ref. 15). Figures 4-5 and 4-8 present the enthalpy data for the wedge and cone respectively. For blunted cones, the THERMO routine is also used to compute local properties (P, V, I) based upon a pressure distribution presented by Griffith-Lewis and isentropic expansion from the stagnation point to that prescribed local pressure. The isentropic expansion assumes $\gamma = 1.40$. Based upon entropy swallowing work by Jacobs, Ref. 13, this should be changed to $\gamma < 1.40$. In any case, the equations were programmed to determine heat transfer and estimate flow field properties in the overexpansion region where sharp cone heat transfer is too high. Within this subroutine, recovery enthalpy and reference conditions are computed for use in Eckert's reference enthalpy technique for high speed heat transfer. The computations for recovery factor assume a Prandtl number (PR) of 0.76. Recovery factor is evaluated as Prandtl number to the 1/2 and 1/3 power for laminar and turbulent flow respectively. Reference enthalpy, density and viscosity are computed from Eq. (4) and the following relationships:

for

$I^* < 1300$

$$\mu^* = 0.426 \times 10^{-7} I^{*0.493} \tag{59a}$$

$$\rho = 0.576 \times 10^{-4} I^{*-0.849} P_\delta \tag{60a}$$

for

$I^* > 1300$

$$\mu^* = 0.284 \times 10^{-6} I^{*0.228} \tag{59b}$$

$$\rho^* = 0.865 \times 10^{-5} I^{*-0.584} P_\delta \tag{60b}$$

The superscript * indicates reference conditions and the subscript δ is used to denote edge of boundary layer property. Viscosity, enthalpy, pressure and density have the usual μ, I, P and ρ symbols.

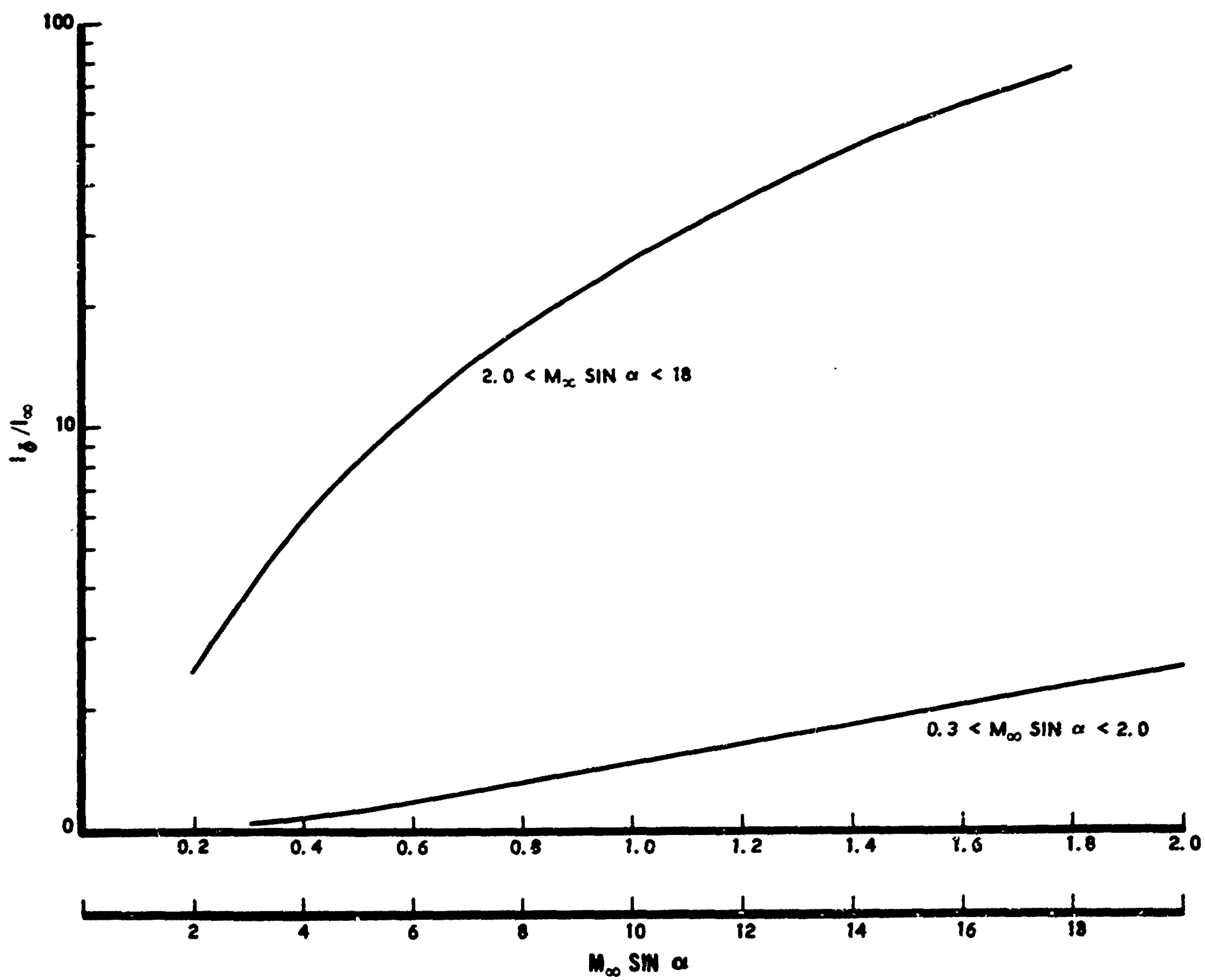


Figure 4-5. Wedge Enthalpy Ratio

71 3613B 82

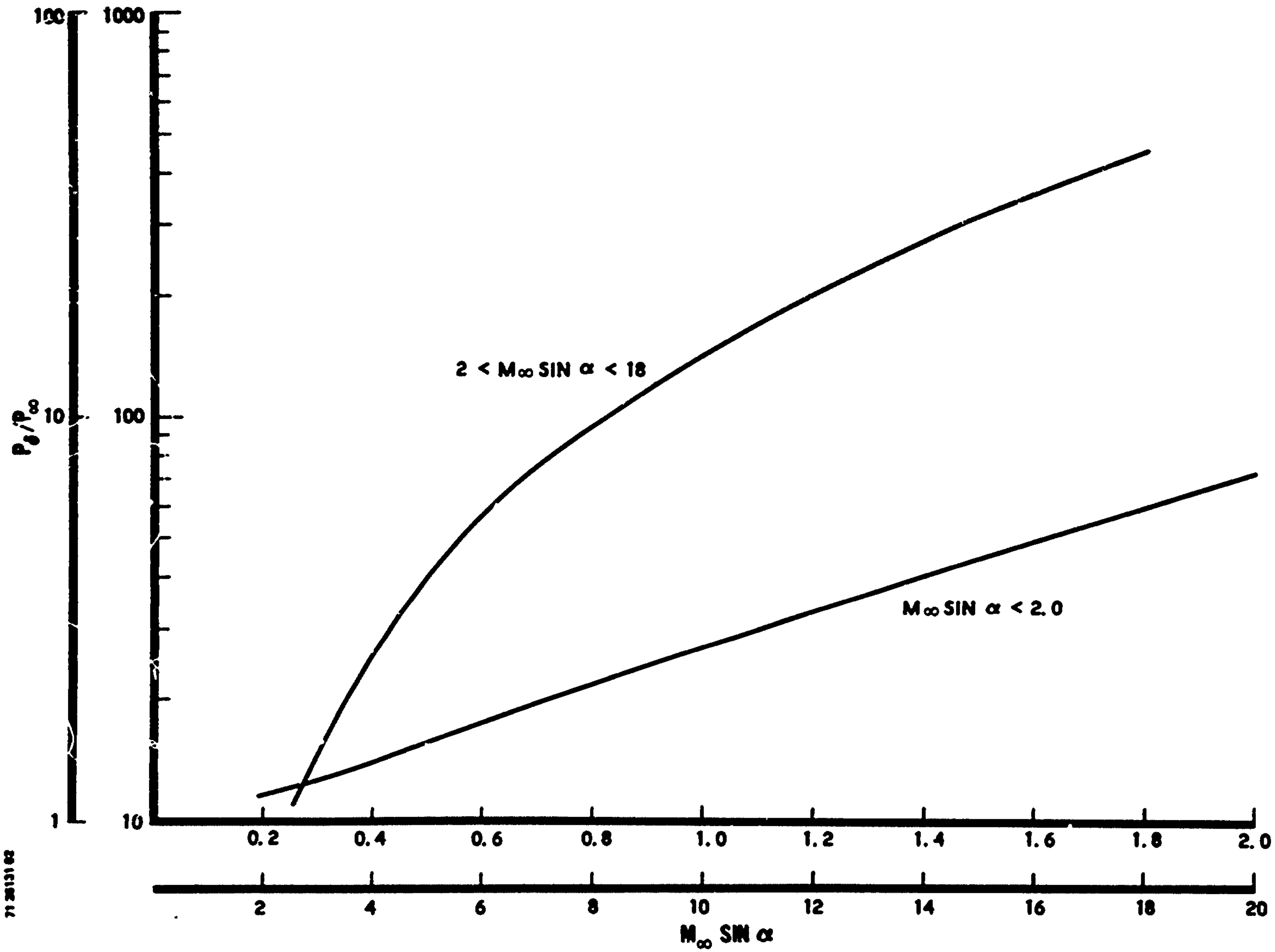


Figure 4-6. Cone Pressure Ratio

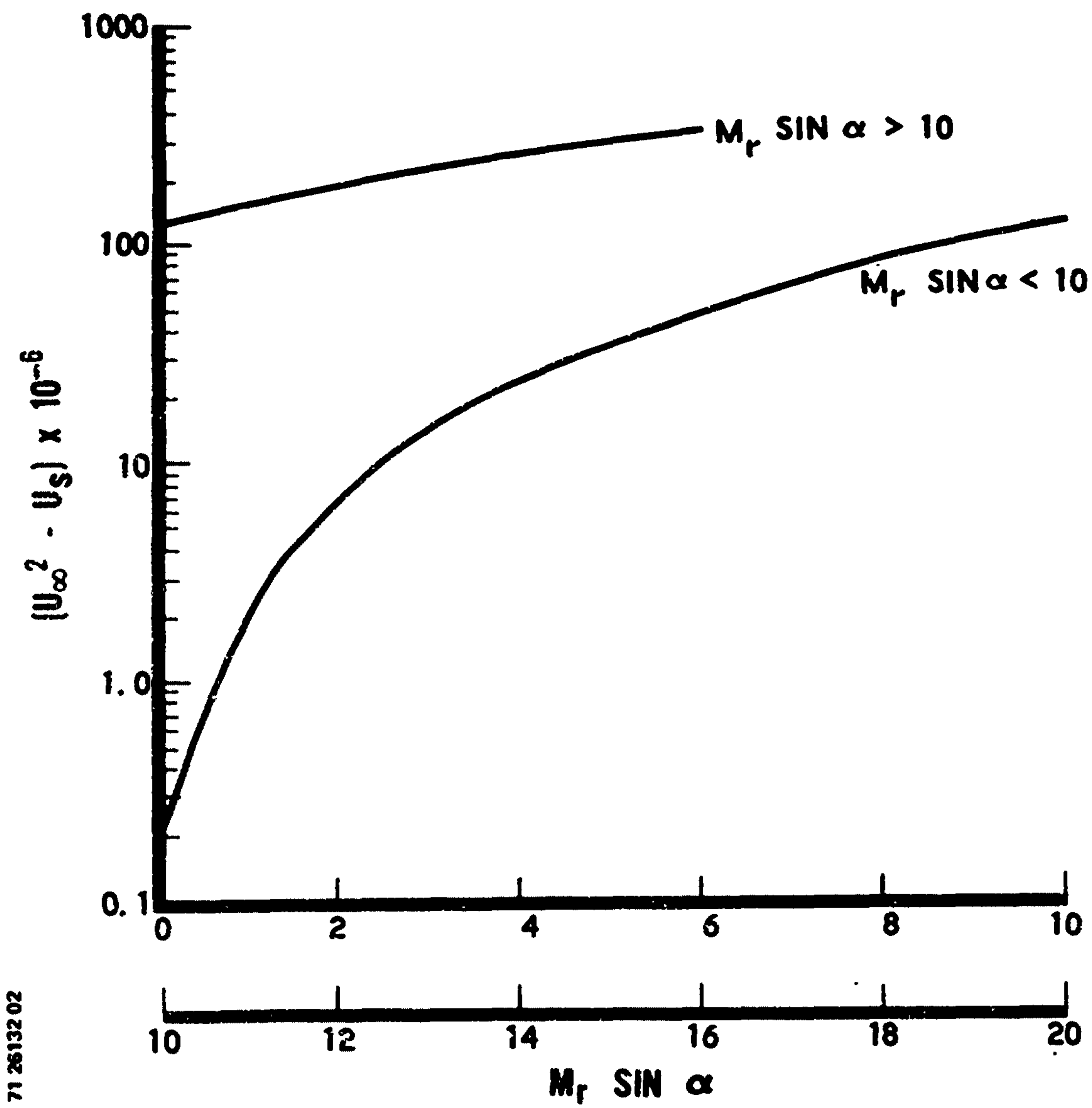


Figure 4-7. Cone Velocity Parameter

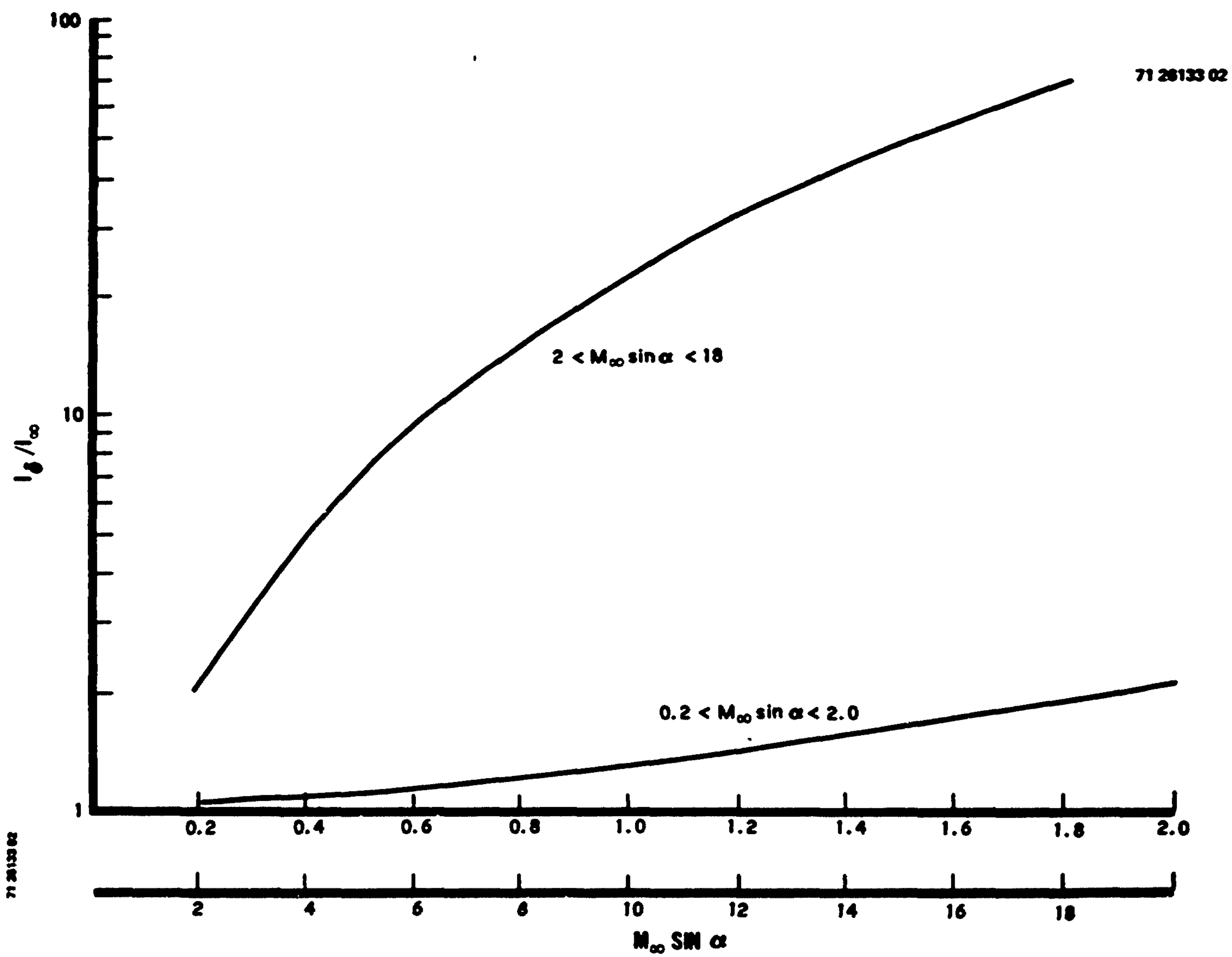


Figure 4-8. Cone Enthalpy Ratio

4.9 DIST

This routine historically modified the input wetted distance to allow for a combined run of laminar and turbulent flow. The input wetted distance was modified to match boundary layer velocity thickness at the Transition Reynolds number. With the computed laminar boundary layer thickness set as the turbulent thickness at transition, an imaginary turbulent wetted length was computed and used in the heat transfer calculations. Experience at Aerospace suggested that the matching of a boundary layer velocity thickness did not yield proper turbulent heat transfer rates. It has been suggested that a forced matching of the momentum thickness would produce more meaningful results. However, at this moment, no such computation is performed within ASAH. The input wetted length is maintained unchanged. The routine is essentially a dummy routine awaiting a reformulation of the mixed wetted length problem. For spheres and cylinders, however, this routine computes their diameter.

4.10 KEMP

This routine is entered whenever ICONF = 3, 4 or 5 (cone, sphere, or cylinder). The stagnation point calculation is performed for sphere and cylinder when ICONF = 4 or 5 respectively. When ICONF = 3 and NOSB = 1 (blunted cone), a stagnation heat transfer is generated for the stagnation point transient conduction problem with nose blunting. For the sphere, the Kemp-Riddel equation is used to compute hot wall heat transfer for high Mach numbers, whereas, the Sibulkin equation is utilized at low supersonic speeds (Mach number less than 3.0). Equations (33) and (34) compute the corresponding $\dot{Q}$ for spheres.

Note that internally the machine program computes heating on a per hour basis. The printout however shows the heat transfer rate in the more useful per second basis.

Eggers, Hansen and Cunningham (Ref. 16) provide the following relationship for the convective heat transfer to an unswept cylinder

$$\dot{Q}_{cyl} = 0.9319\ \dot{Q}_{sphere} \tag{61}$$

Lees (Ref. 18) suggests $(2)^{-1/2}$ as the proportionality constant. The Eggers value is used to obtain conservative heat rates.

The following equations are employed within the ASAH program when considering a cylinder at a prescribed sweep angle geometry (Λ) and angle of attack (α) (useful when modeling the blunted leading edge of a supersonic aircraft for example)

$$ANG = \left[\cos(\alpha)\right]\left[\sin(\Lambda)\right] \qquad (62)$$

$$S = \mathrm{Tan}^{-1} \frac{ANG}{\sqrt{1 - ANG^2}} \qquad (63)$$

$$\dot{Q}_{cyl} = 0.931\ \dot{Q}_{sphere} (\cos S)^{3/2} \qquad (64)$$

4.11 ECKERT

This subroutine computes convective heat transfer for the flat plate, cone, wedge, and blunted cone configurations. Sphere and cylinder configuration flags shift the calculation to subroutine KEMP.

For the blunted cone configuration, the program computes laminar heat transfer convective coefficient based upon the Griffith-Lewis correlation (Ref. 9). This correlation uses the nondimensionalized wetted distance as the entry parameter and interpolates from a heating table carried in the program. [Eq. (18)]

$$Z = \frac{\theta^2 \cos\theta\ X}{0.802\ R}$$

If $Z > 10$ the program assumes bluntness effects have been dissipated by distance, and the program reverts to sharp cone computations.

Under turbulent heating conditions, the blunted cone heating is computed using normal shock-isentropic expansion property data generated in the THERMO subroutine, followed by the Schultz-Grunow turbulent skin friction equation.

For both the sharp cone and sharp wedge configurations this subroutine sequentially computes the following array of terms: skin friction, Stanton number, convective

heat transfer based upon the Blasius and Schultz-Grunow relationship for laminar and turbulent heating respectively. Implicit within this technique to determine heat transfer is the assumption that a modified Reynolds analogy is valid for both laminar and turbulent flow conditions, and that for high speed flow, Eckert's reference enthalpy parameter still correlates the skin friction data. Under extreme flight velocities, the reference enthalpy becomes a more significant measure (relative to reference temperature) of the energy within a flow stream as dissociation and ionization phenomenon affect the stream temperature. The following array of equations reflects this modeling to determine convective heat transfer. [Eq. (2a)]

$$Re^* = \rho^* U_\delta X'/\mu^*$$

For laminar flow conditions [Eq. (2)]

$$Cf = 0.664/(Re^*)^{1/2}$$

For turbulent flow conditions [Eq. (12)]

$$Cf = \frac{0.370}{[\ell n\ Re^*]^{2.584}}$$

Assuming modified Reynolds analogy [Eq. (11)]

$$St = \frac{Cf}{2}\ (Pr)^{-2/3}$$

Computing the convective coefficient and then the convection heating:

$$H_I = St\ U_\delta\ \rho_\delta\ (g)\ (3600) \tag{65}$$

$$\dot{Q}_{conv} = H_I\ (I_r - I_w) \tag{66}$$

Once again the superscript * refers to reference enthalpy conditions and Re, Cf, St, H_I, and $\dot{Q}_{conv}$ are Reynolds number, friction coefficient, Stanton number, convective coefficient based on enthalpy, and convection heating respectively. Note that the internal units for convective heating are Btu/ft^2-hr.

For the cone heating, both the film coefficient and the heat transfer parameter are multiplied by a transformation factor (from Eckert, Ref. 5) to allow for the thinning of the boundary layer as the cone surface area increases as a function of axial length. These transformation terms are $\sqrt{3}$ and $(2)^{0.234}$ for laminar and turbulent flow respectively.

4.12 HANKEY-SONIC POINT SUBROUTINE

The HANKEY subroutine requires an input of ICHK = 1 and radius (RADIUS) of the nose. Calculations at the sonic point are completed for either laminar or turbulent flow. Transition criteria within this routine are based upon either an input transition altitude or a Reynolds number based upon free stream properties and nose radius. For laminar heating the following equation is employed

$$\dot{Q}_{sonic} = (10,800)\left(\frac{\rho_\infty}{\rho_{sl}}\right)^{1/2}\left(\frac{U_\infty}{26,000}\right)^{3.15}\left(\frac{I_s - I_w}{I_s - 129.5}\right)\left(\frac{1}{R}\right)^{1/2} \tag{67a}$$

For turbulent heating at the sonic point

$$\dot{Q}_{sonic} = 5.0\left(\frac{\rho_\infty}{\rho_{sl}}\right)^{0.8}\left(\frac{U_\infty}{1000}\right)^{3}\left(1 - \frac{I_w}{I_s}\right)\left(\frac{1}{R}\right)^{0.2} \tag{67b}$$

The heating rate that is provided as a boundary condition for the transient condition solution is based upon an hourly rate.

$$\dot{Q}_{conv} = 3600\ \dot{Q}_{sonic} \tag{68}$$

4.13 GETCPK

The GETCPK subroutine provides an interpolation subroutine to determine specific heat and conductivity as a function of temperature. The flag needed to direct the calculation is KFUNT. The following code delineates the options on specific heat and conductivity.

KFUNT = 1	specific heat and conductivity are both functions of temperature.
KFUNT = 2	conductivity is a function of temperature; specific heat is constant.
KFUNT = 3	specific heat is a function of temperature; conductivity is constant.
KFUNT = 4	both specific heat and conductivity are constants.
5 ≤ KFUNT ≤ 10	tabular input of specific heat, conductivity and density versus temperature. KFUNT in this case is the number of points describing the temperature curve, with an equal number describing ρ, Cp, K

In general (1 ≤ KFUNT ≤ 4) the specific heat and conductivity are written as power series expansions of temperature. The general form for the equation is

$$C_p \text{ or } K = C_0 + C_1T + C_2T^2 + C_3T^3 \tag{69}$$

with C_0, C_1, C_2, and C_3 being input constants. The values of these constants are assumed zero if omitted in the input. The specific heat and conductivity (and density if KFUNT ≥ 5) for a given time increment are computed for the nodal temperature distribution of the previous time interval.

4.14 ENERGY

This subroutine performs the energy balance at the flow field/material interface. It accounts for high speed convection to the vehicle surface, corrects for current vehicle surface temperature, and for energy radiated away to a zero degree space environment [Eq. (40)]. Internal conduction is evaluated in subroutine ONED.

$$\dot{Q}NET = Cq\ Q_i - \left[(0.174)\,\epsilon\left(\frac{T_w}{100}\right)^4\right]$$

$$Q_i = Q_{CONV}$$

The symbol Cq may be interpreted as an adjustment parameter that modifies the internally computed convective heating. For example, under turbulent flow conic heating conditions, some experts do not utilize the $(2)^{1/5}$ correction factor resulting from the Mangler Conic three-dimensional transformation. If one desires to reproduce this effect within ASAH, a $Cq = \frac{1}{(2)^{1/5}}$ may be used to cancel the programmed turbulent transformation factor for the cone configuration. Another example would be to use Cq as a factor of safety on the predicted heat transfer rates. Factors ranging from 1.2 to 1.5 have been used in the conservative design of reentry heatshields.

4.15 CONSTQ

If NPQ(NPGAS internally) is set equal to a positive number, the CONSTQ subroutine is called from the Main Program. This subroutine provides a heating input based upon a tabular input, and the program bypasses its normal convective heating calculations. If NPQ is a number greater than 2, a linear interpolation of the QI table is completed and set into the convective heating program slot. If NPQ is less than 2, the convective heating term is set equal to QSOLAR, an input constant. The program next computes net heating by subtracting the radiation term, based upon current wall temperature, from the prescribed input cold-wall heating component. This option is useful if the program is to be used to solve the Fourier conduction equation only, bypassing the normal aerodynamic heating routines.

4.16 ONED

This particular subroutine solves the transient one-dimensional conduction problem using the finite-difference form of Fourier's conduction equation. Utilizing subroutine GETCPK, density, conductivity and specific heat for a given point may be obtained as a function of current nodal temperature. The subroutine is programmed such that heat energy may be removed from the back face of the insulative system by an input convective coefficient. Within the system, conduction is the only mode of heat transfer; internal radiation and heat generation are not allowed.

For a thin-skin (one nodal point), the following equation is used to predict temperature change.

$$T_{J+1} = \left(\frac{\Delta t}{\rho a_i \; Cp \; 25.0}\right)\left(\frac{\underline{QNET}}{12}\right) + (HINSD)\left(\frac{TINSD - T_J}{12}\right) + T_J \tag{70}$$

The net energy into the system (QNET) is computed in the ENERGY subroutine. The heat energy removed from the system is computed from an input convective coefficient (HINSD) and an input heat sink temperature (TINSD). Density, specific heat, nodal thickness and temperature are represented by ρ, Cp, a_i and T respectively. The subscript J refers to conditions at the beginning of time interval Δt, whereas the subscript J+1 refers to the temperature at the end of the time interval.

For conduction problems that are divided into more than one nodal point, the following array of difference equations are employed.

Aerodynamic nodal point:

$$T_{n(J+1)} = A1 \left\{ \frac{\underline{QNET}}{12} + \frac{T_{n+1} - T_n}{0.5\left[\frac{a_n}{K_n} + \frac{a_{n+1}}{K_{n+1}}\right]} \right\} \text{ evaluated at J} \tag{40}$$

Interior nodal point:

$$T_{n(J+1)} = A_1 \left\{ \frac{T_{n-1} + T_n}{0.5\left[\frac{a_{n-1}}{K_{n-1}} + \frac{a_n}{K_n}\right]} + \frac{T_{n+1} - T_n}{0.5\left[\frac{a_{n+1}}{K_{n+1}} + \frac{a_n}{K_n}\right]} \right\} + T_{n,J} \tag{39}$$

Internal exposed nodal point:

$$T_{n(J+1)} = A_1 \left\{ \frac{T_{n-1} - T_n}{0.5\left[\frac{a_{n-1}}{K_{n-1}} + \frac{a_n}{K_n}\right]} + HINSD\left[\frac{\underline{TINSD - TN}}{12}\right] \right\} + T_{n,J} \tag{71}$$

The above equations were written around the n^{th} element and were centrally differenced in the interior points. In this case, the subscripts J and J+1 refer to the temperature at the beginning and ending of the calculation time interval (Δt).

The other symbols have the same meaning as previously, with the addition of K for conductivity. A1 represents the heat storage term for the n^{th} element and is defined below.

$$A1 = \frac{\Delta t}{25.0\,\rho_n a_n Cp_n} \tag{41}$$

For the situation where convection to an internal sink is specified (HINSD and TINSD), the internal heat flux is integrated numerically so that it can be printed out at a later time.

4.17 ABLATE

If an ablation temperature is provided in the input, the control routine calls the ablation subroutine. This routine uses a simple effective heat of ablation thermal model to estimate surface recession for an ablative heatshield system. If net heat flux into the system is positive, and if the temperature of the aerodynamically exposed nodal point is equal to the prescribed ablation temperature, this subroutine computes an effective heat of ablation [Eq. (46)]

$$H_{abl} = H_v + \eta\,(I_r - I_w) \tag{45}$$

The H_v and η used in the above equation are input parameters, usually determined from ground test correlations.

For common ablative materials, two values for η are accepted, ETAL and ETAT, the blowing corrections for laminar and turbulent flow respectively. Two equations are available to determine mass loss rate. If the input flag NQI = 0, the ablation rate is determined by subtracting the heat energy conducted inward from the net heat entering the system, and dividing the remaining energy by the materials effective heat of ablation. With NQI > 0, the convective heat transfer, rather than net heat transfer, is used in the mass loss computation. Results are printed in terms of WABDOT and WAB, the mass loss per unit area per calculation time interval and the integrated mass loss per unit area respectively.

For materials that are sensitive to stagnation pressure, the user has the option of degrading heat of ablation at a threshold value of pressure. At that pressure (input as PSTMAX) the program bases heat of ablation on a second set of ablation parameters

(HVPSTM, ETALP, ETATP), usually reflecting a material's inferior performance under a high stagnation pressure environment.

4.18 ABLUNT

The ABLUNT subroutine option is used to predict nose radius change for a sphere/cone configuration. The subroutine is entered by assigning NOSB = 1, and specifying a radius (RADIUS) and an ablation temperature (TWALL) in the input array. The blunting model assumes spherical geometry in the nose region, and uses an effective heat of ablation to predict surface recession. Laminar or turbulent heating is used at either the sonic point or the sphere/cone tangency point. Detailed thermodynamic computations are performed at the stagnation point and engineering correlations are employed to estimate heating at either the sonic or tangency position. The sonic calculation is performed if ISTETS = 0; tangency computations are completed if ISTETS = 1.

Transition may be controlled by specification of a transition Reynolds number or by stipulation of a transition altitude or McCauley criteria (Ref. 2) if neither is specified. With respect to transition, Reynolds number is evaluated at the nose sonic point using sonic point flow field properties (valid only when freestream Mach number is hypersonic), and current nose radius. The flow field properties are evaluated as follows:

$$P_\delta = P_{sonic} = 1160\ P_s \tag{72}$$

$$U_\delta = U_{sonic} = 0.301\ U_\infty \tag{73}$$

$$\rho_\delta = \rho_{sonic} = 7.24\ \rho_\infty \tag{74}$$

$$\mu_\delta = \mu_{sonic} = 1.511 \times 10^{-10}\ U_\infty \tag{75}$$

$$I_\delta = I_{sonic} = (I_s - U^2_{sonic})/5.01 \times 10^4 \tag{76}$$

The Kemp-Ridell equation, discussed earlier, is used to evaluate stagnation heating. Sonic point heating is evaluated using a $(\cos\theta)^{3/2}$ relationship for laminar conditions, and Hankey's equation for turbulent conditions. These relationships are

reproduced below

$$\text{LAMINAR } \dot{Q} = 0.637\ \dot{Q}_s \tag{77}$$

(assumes $\gamma = 1.2$)

$$\text{TURBULENT } \dot{Q} = \frac{5.0}{(R)^{0.2}} \left(\frac{\rho_\infty}{\rho_{sl}}\right)^{0.8} \left(\frac{U_\infty}{1000}\right)^{0.3} \left(1 - \frac{I_w}{I_s}\right) \tag{78}$$

For laminar conditions, using the Stetson option, the heating is evaluated at the sphere/cone tangency point. Figure 4-9, due to Stetson, shows the tangent to stagnation heating ratio for laminar flow conditions. For turbulent conditions, independent of option, computations are always performed at the sonic point.

It should be noted that the conduction problem is solved at only one station (stagnation point) when the blunting option is exercised. The program tacitly assumes that when the ablatic . temperature is attained at the stagnation point, it is simultaneously achieved at either the sonic or tangent points.

Since it has become evident that material thermal performance is degraded at high pressure and/or high shear, this subroutine has been extended to compute effective heat of ablation in a manner that admits variable material performance as a function of external environment. The following equation defines effective heat of ablation in the nose blunting subroutine, using ΔI as $(I_r - I_w)$.

$$H_{abl} = (X_1)(FV_1)\left[HV_1 + \eta_1 \Delta I\right] + (X_1)(1 - FV_1)\left[HF + \eta_1 \Delta I\right] + (1 - X_1)(FV_2)\left[HV_2 + \eta_2 \Delta I\right] \tag{79}$$

The subscripts 1 and 2 refer to the constituents of a two component material (for example the silica and phenolic of OTWR). The weight fraction of component one is designated X_1. FV represents the fraction of material vaporized and is input as a tabular function of pressure and shear. The program user may adjust material performance by operating on the fraction vaporized term. The terms, HV, HF,

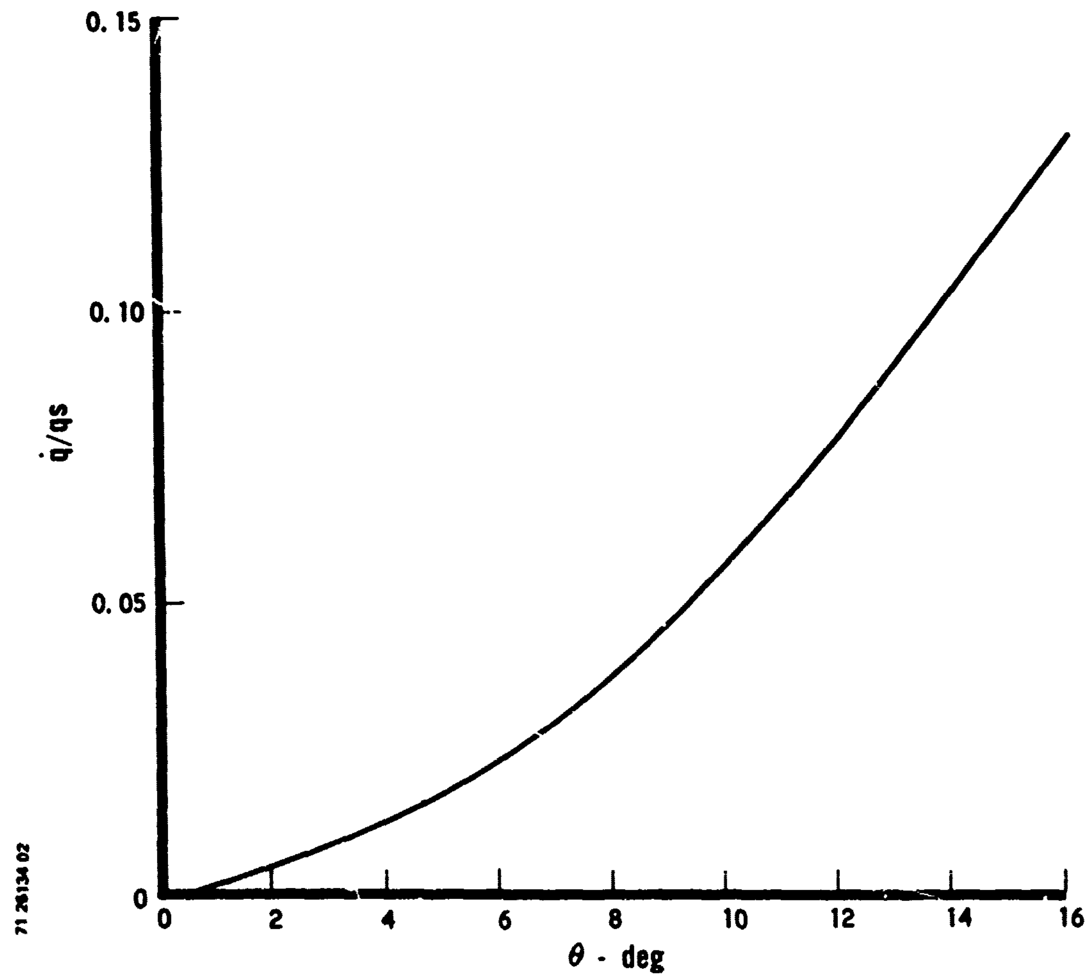


Figure 4-9. STETSON - Tangency to Cone Laminar Heating Ratio

and η refer to heat of vaporization, heat of fusion and blowing factor. If FV and X_1 are not input, the program internally assumes FV = 1 and $X_1 = 1$. When this is the case, the heat of ablation equation in the nose blunting option reverts to the simplified form used in the ablation subroutine Eq. (45).

When using the nose blunting option, the printout has the following meaning. Sonic point pressure, velocity and enthalpy are termed PSHK, VSHK and SHKI respectively. The stagnation heating, stagnation pressure and temperature distribution are printed in their normal array. The Reynolds number displayed is based upon sonic point properties and current nose radius. The stagnation point heat of ablation is termed HOABL.

Additionally, the changing radius and stagnation recession are called RADIUS and F respectively. Further the heat rate at the sonic point (tangent point when "Stetson" option is used), the shear (sonic or tangent points) and the minimum heat of ablation (at sonic or tangent point) are also printed out and are called QSONIC, SHEAR and HOABLM respectively.

With regard to degradation, the program will independently compute a degraded heat of ablation for both pressure and shear conditions, and will use the smaller of the calculated values to determine surface recession.

4.19 PRINT

The PRINT subroutine, called for all normal printing, defines those internal quantities that will be printed out at those times specified by the WRITE input variable. This subroutine has no engineering interest except that several closed-end calculations are performed at this juncture. Free-stream dynamic pressure, shear and stagnation pressure are computed at this point. The following equations are utilized.

$$\mathrm{DYNP} = \rho_\infty U_\infty^2 / 2 \tag{80}$$

$$\mathrm{SHEAR} = \frac{0.025\,\dot{Q}_i\, U_\delta}{(I_r - I_w) * 3600} \;*\mathrm{QCONS} \tag{81}$$

QCONS is an input multiplicative factor used to adjust shear for various reasons (proturberances for example).

Stagnation pressure is computed assuming $\gamma = 1.4$. Units for stagnation pressure are atmospheres.

$$P_{STAG} = (1.20\, M_{\infty}^2)^{3.5} \left[\frac{6}{(7M_{\infty}^2 - 1)} \right]^{2.5} \rho_{\infty}/2116 \tag{82}$$

The variables printed are: TIME, ALTITUDE, U_{∞}, angle of attack (α), Q_i, H_i, QNET, Reynolds number, time step, Δ weight ablated per time step, total weight ablated, I_r, I_w, Q_{inside}, transition Reynolds number, heat of ablation, temperature of ablation, P_{δ}, dynamic pressure, U_{δ}, I_{δ}, I*, total Q, Shear, P_s (atmospheres), P_s (PSF), emissivity, and Δ weight ablated per second. The calculation interval number is printed on the tear strip.

In addition, if the nose blunting option is used, the radius (in feet and inches), estimated nose recession, Q_{sonic}, shear at nose, and heat of ablation at surface are printed.

All this is followed by a list of temperatures at each node.

4.20 GECPK2

This is a two-dimensional version of GETCPK, for finding Cp, K, ρ. Recent modifications now enable the engineer, as an option, to input these as tabular functions of temperature, as described under GETCPK.

4.21 TWOD

The two-dimensional conduction routine, similar to ONED, is discussed under 2.4.2.

4.22 PRINTH

PRINTH is used to print the Hollerith name of the type of configuration used.

4.23 FINTRP

This is an interpolation routine, used for all linear interpolation except that for finding ALT, VINF, AOFA, HINSD, and TINSD in subroutine TRAJ.

4.24 AERSOP

Prints the large Aerospace Corporation symbol at the beginning of each run.

SECTION V

INPUT DESCRIPTION

Input to this program is accomplished by use of IBM supplied utility called NAMELIST (Ref. 17). The data are read from input cards, in subroutine INPUT, and placed on an auxiliary storage device to enable the listing of separate data cases.

Below (Section 5.1) is a description of the method of entry for such items as; Title Cards, Variables, and End of Data cards.

Following this (Section 5.2) is an alphabetic list of variables used in one-dimensional problems, followed by those used for two-dimensional problems.

Section 5.3 then shows typical input examples so that the reader may see how all these tie together.

5.1 DESCRIPTION OF METHOD OF INPUT

5.1.1 Title Card(s)

See "TITLE" in one-dimensional inputs section. (5.2.1)

5.1.2 Variable Entry

5.1.2.1 One-Dimensional

To begin data entry, "&BD06ØS" is entered in columns 2 - 8, followed by at least one blank. The variables and their values then follow, with "=" between the variable and its corresponding values, and a comma (,) following each value. The values corresponding to variable names beginning with any of the letters, I, J, K, L, M, or N, are not entered with a decimal point although one is not necessary for other numbers which have no fractional part. If two or more consecutive values are the same, enter a fixed point number telling how many, followed by an asterisk (*) and the value. Following is an example of entry:

EXAMPLE: A1 = 30., 25., 25., 25., 25., 30.7

Preceding page blank

The data card would look like this:

A1 = 30, 4*25, 30.7, or A1 = 30., A1(2) = 4*25, A1(6) = 30.7,

NOTE: When entering two-dimensional arrays, such as CPTBL (if the element numbers are not specified) the first element number (subscript) varies the most rapidly, i.e., for CPTBL they would vary as: (1,1), (2,1), (3,1)...(I,1), (1,2), (2,2)...(I,2)......(I,J)

It would be well to note, also at this point, that once a value is entered for a given variable, that value will remain in the computer memory, until changed. Should the same value be entered more than once within one data case, however, the last value entered will be used. This particular feature makes this program particularly applicable to running parametric studies.

A "&END" entered in columns 2-5 terminates the data case.

5.1.2.2 Two-Dimensional

For two-dimensional problems the AA, BB, CP2, DK, T2, NARØW, and NBCØL values follow a "&TWØDIM" entered beginning in column 2. As above, equal signs and commas separate the variables and values and a "&END" in columns 2-5 terminates each data case. For multiple cases a "&BD06ØS" card and a "&END" must precede each "&TWØDIM" card.

5.1.2.3 End of Data

To end each set of cases place, on the last card, "ENDDATA" beginning in column 2 with no data following it. Should a comment follow, this comment must not begin before column 9. The card images for each set of cases will be printed preceding the processing of each case. To terminate all processing, place two "ENDDATA" cards together. For further clarification, see Section 5.3 of this writeup and refer to the bottom of Aerospace Form 3703, "4 FIELD NAMELIST INPUT." For easiest data entry for the novice, Aerospace Form 2944, "80 Column FREE Keypunch Form" may be used. However, for orderly results put variable names in columns 2, 22, 42, or 62 as indicated on Form 3703.

5.2 LIST OF INPUT VARIABLES (ALPHABETIC)

5.2.1 One-dimensional Problems (Begin with "&BD06ØS")

The equation for computing the conductivity [Btu/(ft °F hr)] for node J, is as follows:

$K_J = A1_J + A2_J T + A3_J T^2 + A4_J T^3$ where T is temperature [°R]
or entered separately as:

A1(J)[1]	Corresponds to $A1_J$ above
A2(J)	Corresponds to $A2_J$ above
A3(J)	Corresponds to $A3_J$ above
A4(J)	Corresponds to $A4_J$ above
ALT(N)[2]	Altitude [ft] corresponding to TIME(N)
ALTCM(NT)[3]	Altitude [cm] defining a new atmosphere (see TFB)
ANGLE	Cone or wedge half angle [degrees]
AØFA(N)	Angle of attack [degrees] corresponding to TIME(N)

The equation for computing the specific heat [Btu/lb °F] for node J; the equation used is:

$$CP_J = B1_J + B2_J T + B3_J T^2 + B4_J T^3$$

where T is temperature (°R)
or entered separately as:

B1(J)	Corresponds to $B1_J$ above
B2(J)	Corresponds to $B2_J$ above
B3(J)	Corresponds to $B3_J$ above
B4(J)	Corresponds to $B4_J$ above

[1] (J) (is less than or equal NSEG) refers to nodal distance dependent variables.

[2] (N) (cannot be greater than NTRAJ) refers to time dependent variables.

[3] (NT) (cannot be greater than TRVRSL) refers to altitude dependent atmospheric variables.

J and N will be used in this same context in the rest of this writeup.

CONPR	Limiting cone pressure ratio for Prandtl Meyer Expansion. The program will not lower the pressure ratio below this constraint.
CPTBL(I, J)[4]	Specific heat of segment J corresponding to TEMTBL(I, J) [Btu/lb °F]
DELTA	Time step size [sec]. This is no longer a fixed increment. After program start-up the maximum stable increment will be used. A number must be entered. DELTA will not exceed WRITE, however.
EKTBL(I, J)[4]	Thermal conductivity of segment J corresponding to TEMTBL(I, J) [Btu/(ft °F hr)]
EL(J)	Thickness [inches] of the Jth segment
EMIS	Surface emissivity
EMTAB(L)[5]	Emissivity table, each value corresponding to TWTAB(L)
ETAL ETAT	Eta laminar and Eta turbulent, used in the calculation of the heat of ablation (see HV), a blowing correction
ETAL2 ETAT2	Same as ETAL or ETAT, but for the second component within a composite (see HV2)
ETALP ETATP	Same as ETAL and ETAT, but used only when PSTAG is greater than PSTMAX

[4](I)Max. of KFUNT, minimum of 5, refers to individual TEMTBL vs RHOTBL, EKTBL, CPTBL points for material properties interpolation.

NOTE: Only the properties of the first segment of a given material need be entered for the TEMTBL vs RHOTBL, EKTBL, CPTBL option, provided all were initialized to zero, which they are for the first data case.

[5](L) maximum of 20, refers to individual TMTAB vs EMTAB values (wall temperature °R vs emissivity), when emissivity is variable. (See NEMIS)

FV1S(K)[6]	Fraction vaporized component 1 (shear dependent) of two component composite material (see XI).
FV2S(K)	Fraction vaporized component 2 (shear dependent) entered as tables (max. of 10 points) vs SHRTBL(K).
FV1P(K)	Fraction vaporized component 1 (pressure dependent).
FV2P(K)	Fraction vaporized component 2 (pressure dependent) entered as tables (max. 10 points) vs STGTBL(K).
HEDGE(N)	Input table from Entropy Swallowing Program when INPCW $\geq$ 1, corresponds to SHKI (I_6 internally)
HF	Heat of fusion for component 1, if not all vaporized (see XI) [Btu/lb]
HINSD1(N)	Inside heat transfer at TIME(N) [Btu/hr ft^2 °R]
HØABL	Heat of ablation [Btu/lb]
HRECØV(N)	Corresponds to HSTAG in Entropy Swallowing Program (table input vs TIME1 of INPCW $\geq$ 1 and corresponds to RECI inside this program)
HV	Heat of vaporization for first component of a heatshield [Btu/lb]
HV2	Heat of ablation for second component of a composite heatshield
HVPSTM	HV for PSTAG greater than PSTMAX [Btu/lb]
ICØNF	= 1 Flat plate calculation = 2 Wedge = 3 Cone or blunt cone (see NØSB) = 4 Sphere (see NØSB) = 5 Cylinder

[6](K) Max. of 10 refers to pressure dependent variables.

IHANK	Hankey sonic point flag if ICØNF = 4 Set to 1 to use, set to 0 to block usage. RADIUS must also be specified.
INPCW	If greater than zero, signals that cold wall inputs (TIME, QINT, HEDGE, UEDGE, PEDGE, and HRECØV) will be processed and used for computing QSUBI instead of Aero-Heating. The value of INPCW tells how many points in the above tables (maximum of 50).
ISTETS	For Ste' on type laminar nose blunting heating when positive.
KASE	Case identification number ($-99 \leq$ KASE ≤ 999)
KFUNT	Determines if K and/or C_p are functions of temperature = 1 both are functions of temperature = 2 K only is a function of temperature = 3 C_p only is a function of temperature = 4 neither is a function of temper ıre, but both are constant. (see A1 a.ı B1) $\geq$ 5 is the number of points (no more than 10) in ρ, K, C_p vs temperature tables, from which material properties are interpolated, for each segment at their particular segment temperatures (see TEMTBL, RHØTBL, EKTBL, CPTBL)
NDBUG	Debugging flag, gives a print each time through one of the routines listed for certain sections of each = 1 PRINT, ABLATE, GETCPK when $5 \leq$ KFUNT ≤ 10 $\geq$ 1 ABLUNT, ABLATE, INPUT, KEMP, PRINT

	$\geq$ 1, $\neq$ 3 TRAJ
	= 2 ABLUNT
	$\geq$ 2 DIST, GECPK2
	= 3 PMF, ATMØS
	= 4 SHOCK
	= 5 THERMØ
	= 8 HANKEY
	= 14 ENERGY/ENCO
	$\geq$ 14 TWØD
NDIM	= 1 for 1-dimensional problems
	= 2 for 2-dimensional problems
NEMIS	Normally zero, used as number of entries in variable emissivity table (EMTAB vs TWTAB), maximum = 20
NØPSTG	No. of entries in FV1P and FV2P vs STGTBL tables, maximum = 10
NØPSHR	No. of entries in FV1S and FV2S vs SHRTBL tables, maximum = 10
NØSB	Used with ICØNF = 3 or 4, Set = 1 for Nose Blunting. RADIUS and TWALL must be entered. Set = 0 if no blunting.
NPQ	Set to one of the positive numbers below for NON-AERO Heating Option
	= 1 QSUBI = QSØLAR (constant value that must be input.)
	= 2 $\leq$ NPQ $\leq$ 100, QSUBI will be interpolated from the QI vs TIME tables from NPQ (number of) values.

NPQS	Normally zero, used as number of entries in QSTAR vs PSTAGN tables (10 maximum)
NPU	Punch flag - Set positive to punch cards that serve as input for Char Ablator Program. These punched inputs are:

TIME	=	TSTØR
QSUBI	=	QØ
PS	=	PØ
RECI	=	HSTAB
SHKI	=	HLTAB
VSHK	=	UETAB

Since a value is stored for each of the above variables at time of print, it is recommended that

$$\text{WRITE} \geq \frac{\text{STOP}}{\text{50-NSEG}} \text{ to be ablated}$$

so that NTMP, NTME, NØP, NHS, NUE, NHLØC (No. table values) will be less than 50.

NQI	When set positive will cause the ablation routine to use QSUBI in place of QNET in the weight ablated (WABDØT) calculation. If NQI = 0, QNET will be used in the calculation.
NSEG	Number of segments in heated material, ($1 \leq$ NSEG ≤ 40)
NTRAJ	Number of points in trajectory table ($1 \leq$ NTRAJ ≤ 100) for TIME vs VINF, vs ALT, vs AØFA, QCØNS, TINSD, HINSD, etc.
PEDGE(N)	Table input from Entropy Swallowing Program when INPCW ≤ 1 corresponding to PSHK [Atmosphere]
PFB(NT)	Pressure [dynes/cm^2] corresponding to TFB(NT) and ALTCM(NT)

PSTAGN(K)[7]	Table of stagnation pressures corresponding to QSTAR(K) (see NPQS)
PSTMAX	Maximum value for PSTAG before switching from equation 46 to HØABL = HVPSTM x η x $(I_S - I_W)$
QCØNS(N)	Table of multiplying constants vs TIME used in the Aero Heating QSUBI equation. If QCØNS (NTRAJ) = 0, all QCØNS will = 1.
QI(N)	Table values, at Time = TIME(N), for QSUBI when NPQ > 1. [Btu/ft^2-sec]
QINSD(N)	Back wall Q vs TIME(N)
QINT(N)	Cold wall Q from Entropy Swallowing Program (vs TIME) to be used when INPCW $\geq$ 1. This is converted to hot wall Q internal to the program.
QSØLAR	A single value for heating, either above 295274.7 ft (90,000 meters) or if NPQ = 1 below that altitude.
QSTAR(K)	Used as HV, interpolated from PSTAGN(K)
RADIUS	Of sphere, cylinder, or of blunt nose when NOSB = 1 [ft]. Otherwise use XDIST for sphere or cylinder diameter. With NOSB = 0, RADIUS will trigger the Griffith-Lewis Heating for cones.
RHØ(J)	Density of Jth material segment [lb/ft^3]
RHØFB(NT)	Density of atmosphere [gm/cm^3] at ALTCM(NT)
RHØTBL(I, J)	Material density of segment J corresponding to TEMTBL(I, J) [lb/ft^3]
START	Starting time of desired trajectory if other than zero, may be greater than TIME (1) [sec]

[7] K has a maximum value of 10.

STGTBL(K)	Independent parameter in vaporization vs pressure table [atmospheres]
SWEEP	Sweep angle [deg] for cylinder with axis not perpendicular to velocity vector
TABL	Temperature of ablation [°R]; nose blunting requires an additional input, TWALL
TEMP(J)	Initial temperature of node "J" [°R]
TEMTBL(I, J)	Temperature table values (when KFUNT $\geq 5 \geq 10$) from which material properties of segment J will be interpolated [°R]
TFB(NT)	Temperature [°K] at ALTCM(NT)[8] (See TRVRSL)
TITLE	A - 1 to 72 alphanumeric character title which is printed preceding each case. Must begin and end with an apostrophe
TIME(N)	Time corresponding in trajectory to VINF(N), ALT(N)
TINSD1(N)	Temperature inside [°R] at TIME(N)
TRANLT[9]	Transition Altitude [ft] only if TRANR = 0
TRANR	Transition Reynolds number
TRVRSL	No. of points defining new atmosphere (see TFB)
TWALL	Temperature of Ablation if NØSB = 1 [°R] must be input each case or TABL will be used.
TWTAB(L)	Temperature at wall corresponding to EMTAB(L)

[8] If any two consecutive TFB values are zero, the program will use standard atmosphere during that portion of the trajectory when two or more TFB values are positive, an interpolation for TINF, RHØINF, PINF vs ALTCM is accomplished, but only between positive TFB values.

[9] If both TRANLT and TRANR = 0, McCauley Transition Criteria will be used.

UEDGE(N)	Table input from Entropy Swallowing Program when INPCW > 1 corresponding to VSHK internally
VINF(N)	Free-stream velocity at TIME(N) [fps]
WRITE	Print interval for problem [sec]
XDIST	The distance back from the leading edge of a wedge or cone at which temperature profile is desired, or the diameter of sphere or cylinder [ft].
XI	Fractional portion of component 1 of a two component composite ablative material (see ETAT2, ETAL2, HF, HV2, FV1, and FV2) XI taken to be 1, unless otherwise specified.

5.2.2 Two-Dimensional Problems (Begin with "&TWØDIM")

Following the normal one-dimensional inputs, the additional inputs needed for two-dimensional problems (beginning with "&TWØDIM" instead of "&BD06ØS") are:

AA(J)	Segment thickness (in Y direction) corresponding to EL(J) [inches]
BB(M)	Segment width (in X direction) [inches] axial flow direction
CC(J, M)	≥ 1 for conduction into and out of grid point $= 0$ for no conduction, such as perfect insulators
CP2(J)	Corresponds to B1(J), specific heat capacitance[10]
DK(J)	Corresponds to A1(J) conductivity
T2(J, M)	Initial temperature distribution in two-dimensional grid [°R]

[10] Two-dimensional problems are assumed to be homogeneous in the X, (Parallel to surface) (Subscript M) direction, but not necessarily in the Y (in depth) (subscript J). J and M for two-dimensional problems may not exceed 10.

NARØW	Number of rows of AA (Max. = 10)
NBCØL	Number of columns of BB (Max. = 10)

5.3 EXAMPLE INPUTS

Following are examples for inputting both one-dimensional (5.3.1) and two-dimensional (5.3.2) problems. These problems show certain features and uses of the program. The following is a general description of each of the problems.

CASE 1 is a one-dimensional, single segment problem using fixed material properties and deriving its heating from aerodynamic criteria which come from an input trajectory using McCauley transition criteria.

CASE 2 is a one-dimensional, five segment problem, with C_p input as a function of temperature, while ρ and K are held constant. Heating, for this problem, comes from the cold wall input parameters associated with "INPCW = 1". This problem uses altitude for transition criteria.

CASE 3 is a modification of Case 2, and adds another segment of different material to the face of the five segments, thus making it a six segment problem with all the features described in the above paragraph.

CASE 4 shown on the printed listing (as made by the ASAH Program) as Cases 4 and 5, is a two-dimensional problem with segment 2 in slab 2 not present. The "START" feature was employed, allowing the program to start at some time other than at the first time entry. The second "T2" card, it should be noted, supersedes the first "T2" card. The values on the second "T2" card were obtained from the values on the first "T2" card after running the problem for 68.7 seconds. All other parameters are as described in Case 2, above.

The following is a listing made by the ASAH program illustrating both the actual input cards used, and the standard output as given by Subroutine INPUT.

COMPLETE LIST OF ALL INPUT CARDS TO BE PROCESSED

```
SAMPLE 1 AND 2 DIMENSIONAL PROBLEMS - FOR A.S.A.H. MAR.3,'71          
&BD06CS
TITLE=' SAMPLE ONE DIMENSIONAL PROBLEM, ONE SEGMENT, AERO HEAT',
ISTETS=0,AOFA=16*0,HVPSTM=0.,PSTMAX=0.,KASE=1,NDIM=1,
ICONF=3,ANGLE=8.,DELTA=.05,WRITE=10.,NSEG=1,KFUNT=4, NTRAJ=16
EMIS=.85,TABL=10000.,HV=1000.,ETAL=1.,ETAT=1.,
TIME=0.,22.,40.,54.,60.,70.,80.,96.,112.,118.,124.,130.,136.,        5.3.1.1
140.,144.,148.,
ALT=200.,4350.,15500.,19540.,37000.,51570.,69210.,104800.,151270.,
172125.,195210.,220540.,245180.,261700.,277990.,294230.,
VINF=10.,410.,915.,1410.,1690.,1850.,2000.,2630.,4000.,5850.,7550.,
    8690.,9350.,9750.,9970.,10110.,10270.,
A1=60,    B1=.28,     RHO=108.,    EL=.125,     STOP=14.8,   XDIST=.243,
&END          ONE DIM., ONE SLAB, ONE MAT'L
                                                                      END CASE     1
&BD0600S  KASE=2,TITLE(9)=' - W','ITH ','COLD',' WAL','L IN','PUTS',
', 5 ','SEG.',
KFUNT=3,
XDIST=.243,TRANLT=80000.,NTRAJ=17,INPCW=17,
TIME=0.,22.,40.,54.,60.,67.,70.,80.,96.,112.,118.,124.,130.,136.,140.,
     144.,148.,
QINT=.1,.28,1.00,15.9,15.6,11.36,6.42,3.97,2.88,3.43,4.12,3.68,
    2.57,1.57,1.08,.711,.375,
ALT=200.,4350.,15500.,19540.,37000.,47000.,51570.,69210.,104800.,
    151300.,172125.,195210.,220540.,245180.,261700.,277990.,295000.,
PEDGE=.995,.84,.55,.35,.25,.15,.13,.062,.010,.0033,.00131,6.6E-4,
     2.65E-4,9.47E-5,4.23E-5,1.86E-5,7.69E-6,
HRECOV=123.,123.,126.,140.,146.,160.,167.,221.,400.,700.,1093.,1393.,
     1574.,1683.,1753.,1800.,1907.,                                   5.3.1.2
UEDGE=10.,400.,900.,1300.,1550.,1775.,1800.,2400.,3600.,5500.,7149.,
     8320.,9084.,9593.,9802.,9977.,10000.,
HEDGE=123.,123.,120.,130.,165.,165.,165.,170.,190.,210.,242.,240.,
     199.,149.,153.,140.,127.,
TEMP=8*540., TRANLT=47000.,
NSEG=5,
A1=60.,.25,.965,2*75.,
B1=.0,.28,.125,2*.034,B2=.0003,4*.0,
RHC=108.,150.,334.,2*1192.,
EL=.125,.010,.050,2*.25,
WRITE=5.,
&END     NOTE - VARIABLE THERMAL PROPERTIES
                                                                      END CASE     2
```

```
                                                         5.3.1.2.1
&BDJ6GS QCONS=5*.495,
&END    CHANGED QCONS
                                                         END CASE   3

&BDJ6OS  TITLE(15)=', SI','X SE','GMEN','TS  ',   KASE=3,
NSEG=6,
QCCNS=6*1.,
TEMP=6*540.,                                             5.3.1.3
A1=.14,60.,.25,.965,2*75.,
B1=.26,.0,.28,.125,2*.034, B2=.0,.0003,4*.0,
RHO=136.,108.,150.,334.,2*1192.,
EL=.010,.125,.010,.050,2*.25,
EMIS=.15,HV=950.,ETAL=.43,ETAT=.17,TABL=1800.,
&END    NOTE - CHANGED EMIS, HV, TABL, FCT.              END CASE   4

&BDJ6OS  TITLE=' TWO-DIMENSIONAL PROBLEM, BASED ON KASE 3 (1-D)',
KASE=23, NDIM=2,
WRITE=.02,
START= 68.7,                                             END CASE   5
&END
                                                         5.3.2
&TWODIM
AA=.125,.010,.050,2*.25,
BB=.308,.308,
CC=5*1,5*0,1,0,3*1,80*0.,
T2=5*540.,5*0,540,0,3*540,85*0,
NARCW=5,
NRCOL=2,
T2=657,632,577,546,545,5*C,665,0,3*544,85*0,             END CASE   6
&END

ENDDATA                                                            0

END OF INPUT.
   66 CARDS READ
```

SECTION VI

SAMPLE OUTPUTS

Five cases are shown on the following pages. These correspond to the example input cards shown in Section 5.3 of this writeup. Descriptions of these cases are as follows:

6.1 One Dimensional, One Segment, Aerodynamic Heating

6.2 One Dimensional, Five Segments, Non-Aeroheating

6.3 One Dimensional, Five Segments, Non-Aeroheating, Q Correction

6.4 One Dimensional, Six Segments, Non-Aeroheating, No Q Correction

6.5 Two Dimensional, Five Segments x Two Slabs, Non-Aeroheating

Useful comments about the variables which are output, and what they mean may be found in Section 4.19.

6.1 ONE DIMENSIONAL, ONE SEGMENT, AERODYNAMIC HEATING

```
 INPUT CARDS READ, DATA SET      1
SAMPLE 1 AND 2 DIMENSIONAL PRCBLEMS - FOR A.S.A.H. MAR.3,'71
&BC060S
TITLE=' SAMPLE ONE DIMENSIONAL PROBLEM, ONE SEGMENT, AERO HEAT',
ISTETS=0,AOFA=16*C,HVPSTM=0.,PSTMAX=   ,KASE=1,NDIM=1,
ICCNF=3,ANGLE=8.,DELTA=.05,WRITE=10.  EG=1,KFUNT=4, NTRAJ=16
EMIS=.85,TABL=10000.,HV=1000.,ETAL=1.,ETAT=1.,
TIME=0.,22.,40.,54.,60.,7C.,80.,96.,112.,118.,124.,130.,136.,
140.,144.,148.,
ALT=200.,4350.,15500.,19540.,37000.,51570.,69210.,104800.,151270.,
172125.,195210.,220540.,24518C.,261700.,277990.,294230.,
VINF=10.,410.,915.,1410.,1690.,1850.,2000.,2630.,4000.,5850.,7550.,
    8690.,9350.,9750.,997C.,1C110.,10270.,
A1=60,    B1=.28,     RHO=108.,    EL=.125,     STOP=14.8,  XDIST=.243,
&END           ONE DIM., ONE SLAB, ONE MAT'L
KARC =      14,   TO GO =     52

END OF INPUT.
```

```
 SAMPLE ONE DIMENSIONAL PROBLEM, ONE SEGMENT, AERO HEAT
IDENTIFICATION = &BD060S
KASE    1
  CONE
INTEGERS
    ICONF      ICHK     IHANK     INPCW    ISTETS      KASE     KFUNT     NBCOL     NDBUG      NDIM
        3         0         0         0         0         1         4         0         0         1
   NOPSHR    NOPSTG      NOSB    NPGAS       NPQ       NPU       NQI      NSEG     NTRAJ    NOCHEK
        0         0         0         0         0         0         0         1        16         0
       -1
```

SAMPLE ONE DIMENSIONAL PROBLEM, ONE SEGMENT, AERO HEAT

```
   TRANR              TRANLT            ANGLE                SWEEP            XDIST              CONPR
0.0             0.0                0.80000000E 01  0.0                    0.24299997E 00  0.0

    DELTA              WRITE             STOP
  0.49999997E-01     0.10000000E 02    0.14799999E 02

    TABL               HV                ETAL                 ETAT                  EMIS
  0.10000000E 05     0.10000000E 04    0.10000000E 01       0.10000000E 01        0.84999996E 00

J         A1(J)             A2(J)             A3(J)             A4(J)
1  0.60000000E 02  0.0               0.0               0.0

J         B1(J)             B2(J)             B3(J)             B4(J)
1  0.27999997E 00  0.0               0.0               0.0

          RHO(I), I=1,   1
   0.10800000E 03

          EL(I), I=1,   1
   0.12500000E 00

         TEMP(I), I=1,   1
   0.54000000E 03
```

I	TIME	ALT	VINF	AOFA	HINSD	TINSD	QCONS	QI*3600	TEMPW	VAPRES
1	0.0	200.0	10.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
2	22.0	4350.0	410.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
3	40.0	15500.0	915.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
4	54.0	19540.0	1410.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
5	60.0	37000.0	1690.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
6	70.0	51570.0	1850.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
7	80.0	69210.0	2000.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
8	96.0	104800.0	2630.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
9	112.0	151270.0	4000.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
10	118.0	172125.0	5850.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
11	124.0	195210.0	7550.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
12	130.0	220540.0	8690.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
13	136.0	245180.0	9350.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
14	140.0	261700.0	9750.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
15	144.0	277990.0	9970.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
16	146.0	294230.0	10110.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

QSOLAR

```
INITIAL CONDITIONS
     TIME              ALT              VINF              AOFA              QSUBI             HSUBI
  0.0              0.0              0.0              0.0              0.0              0.0
     QNET              REYN             DELTA             WABDOT            WAB               RECI
  0.0              0.0              0.49999997E-01   0.0              0.0              0.0
     WALLI             QSAV             QINSD             TRANR             HOABL             TABL
  0.0              0.0              0.0              0.0              0.0              0.10000000E 05
     PSHK              DYNP             VSHK              SHKI              REFI              QNSUM
  0.0              0.0              0.0              0.0              0.0              0.0
     SHEAR             PS               PSTAG             EMIS              QCONS             WAB/SEC
  0.25098830E 00   0.0              0.0              0.94999996E 00   0.0              0.0

 1ST SEGMENT PRESENT=NSEG2 =   1.   TEMP(I), I=NSEG2,NSEG FOLLOW.
    540.0
INITIAL CONDITIONS

 TRANSITION HAS OCCURRED * * * * * * CONDITIONS BELOW,  FROM UNKNOWN   TO  TURBULNT TRANLT=  0.0

     TIME              ALT              VINF              ACFA              QSUBI             HSUBI
  0.49999997E-01   0.20943181E 03   0.10909090E 02   0.0              -0.24550151E-01  0.49067810E-02
     QNET              REYN             DELTA             WABDOT            WAB               RECI
 -0.59433433E-01   0.11743426E 05   0.49999997E-01   0.0              0.0              0.12338243E 03
     WALLI             QSAV             QINSD             TRANR             HOABL             TABL
  0.12838574E 03   0.0              0.0              0.34916519E 02   0.0              0.10000000E 05
     PSHK              DYNP             VSHK              SHKI              REFI              QNSUM
  0.21002327E 04   0.14725348E 0    0.10909090E 02   0.12338026E 03   0.12588347E 03   0.0
     SHEAR             PS               PSTAG             EMIS              QCONS             WAB/SEC
  0.13382125E-02   0.99254947E 00   0.99261475E 00   0.84999996E 00   0.10000000E 01   0.0

 1ST SEGMENT PRESENT=NSEG2 =   1.   TEMP(I), I=NSEG2,NSEG FOLLOW.
    540.0
STABILITY  CHECK,  DEL, CCN, JAV, WRI, DELTA
          0.09843743        0.25000000          0.02499999        400.00000000         0.02500000       -1

     TIME              ALT              VINF              ACFA              QSUBI             HSUBI
  0.99748621E 01   0.20810216E 04   0.19136111E 03   0.0              -0.19426495E 00  0.41107066E-01
     QNET              REYN             DELTA             WABDOT            WAB               RECI
 -0.22790742E 02   0.19582206E 06   0.24999999E-01   0.0              0.0              0.12244653E 03
     WALLI             QSAV             QINSD             TRANR             HOABL             TABL
  0.12717236E 03   0.0              0.0              0.81685352E 04   0.0              0.10000000E 05
     PSHK              DYNP             VSHK              SHKI              REFI              QNSUM
  0.19618015E 04   0.40E331C4E 02   0.19136111E 03   0.12177901E 03   0.12462253E 03   0.0
     SHEAR             PS               PSTAG             EMIS              QCONS             WAB/SEC
```

```
0.19665718E 00    0.92712736E 00    0.94642693E 00    0.84999996E 00    0.10000000E 01    0.0

1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
   534.9

     TIME              ALT               VINF              ACFA              QSUBI             HSUBI
 0.14824788E 02    0.29964939E 04    0.27954150E 03    0.0              -0.20941544E 00    0.54173905E-01
     QNET              REYN              DELTA             WABDOT            WAB               RECI
-0.24213862E 00    0.27899719E 06    0.24999999E-01    0.0               0.0               0.12242154E 03
     WALLI             QSAV              QINSD             TRANR             HOABL             TABL
 0.12628716E 03    0.0               0.0               0.16886055E 05    0.0               0.10000000E 05
     PSHK              CYNP              VSHK              SHKI              REFI              QNSUM
 0.18969067E 04    0.84804962E 02    0.27954150E 03    0.12099706E 03    0.12395549E 03    0.0
     SHEAR             PS                PSTAG             EMIS              QCONS             WAB/SEC
 0.37859607E 00    0.89645875E 00    0.93653667E 00    0.84999996E 00    0.10000000E 01    0.0

1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
   531.2
```

(NT = 592)

6.2 ONE DIMENSIONAL, FIVE SEGMENTS, NON-AEROHEATING

```
 INPUT CARDS READ, DATA SET      2
&BDJ60S  KASE=2,TITLE(9)=' - W','ITH ','COLD',' WAL','L IN','PUTS',
', 5 ','SEG.',
KFUNT=3,
XDIST=.243,TRANLT=80000.,NTRAJ=17,INPCW=17,
TIME=0.,22.,40.,54.,60.,67.,70.,80.,96.,112.,118.,124.,130.,136.,140.,
     144.,148.,
QINT=.1,.28,1.00,15.9,15.6,11.36,6.42,3.97,2.88,3.43,4.12,3.68,
    2.57,1.57,1.08,.711,.375,
ALT=200.,4350.,15500.,19540.,37000.,47000.,51570.,69210.,104800.,
    151300.,172125.,195210.,220540.,245180.,261700.,277990.,295000.,
PEDGE=.995,.84,.55,.35,.25,.15,.13,.062,.010,.0033,.00131,6.6E-4,
     2.65E-4,9.47E-5,4.23E-5,1.86E-5,7.69E-6,
HRECOV=123.,123.,126.,140.,146.,160.,167.,221.,400.,700.,1093.,1393.,
     1574.,1683.,1753.,1800.,1907.,
UEDGE=10.,400.,900.,1300.,1550.,1775.,1800.,2400.,3600.,5500.,7149.,
     8320.,9084.,9593.,9802.,9977.,10000.,
HEDGE=123.,123.,120.,130.,165.,165.,165.,170.,190.,210.,242.,240.,
     199.,149.,153.,140.,127.,
TEMP=8*540., TRANLT=47000.,
NSEG=5,
A1=60.,.25,.965,2*75.,
B1=.0,.28,.125,2*.034,B2=.0003,4*.0,
RHO=108.,150.,334.,2*1192.,
EL=.125,.010,.050,2*.25,
WRITE=5.,
&END     NOTE - VARIABLE THERMAL PROPERTIES
KARD =     26,   TO GO =    26

END OF INPUT.
```

```
 SAMPLE ONE DIMENSIONAL PROBLEM, - WITH COLD WALL INPUTS, 5 SEG.
ICENTIFICATION = &BD060S
KASE    2
  CCNE
INTEGERS
     ICONF      ICHK     IHANK     INPCW    ISTETS      KASE     KFUNT     NBCOL     NDBUG      NDIM
         3         0         0        17         0         2         3         0         0         1
    NOPSHR    NOPSTG      NOSR     NPGAS       NPQ       NPU       NQI      NSEG     NTRAJ    NOCHEK
         0         0         0         0         0         0         0         5        17         0
         0
```

```
SAMPLE ONE DIMENSIONAL PROBLEM, - WITH COLD WALL INPUTS, 5 SEG.

     TRANR              TRANLT          ANGLE                 SWEEP            XDIST          CONPR
0.0               0.47000000E 05  0.80000000E 01  0.0                     0.24299997E 00  0.0

      DELTA              WRITE               STOP
   0.49999997E-01    0.50000000E 01     0.14799999E 02

      TABL               HV                  ETAL                   ETAT                   EMIS
   0.10000000E 05    0.10000000E 04     0.10000000E 01        0.10000000E 01        0.84999996E 00

J          A1(J)              A2(J)                A3(J)              A4(J)
1  0.65000000E 02  0.0                0.0                0.0
2  0.25000000E 00  0.0                0.0                0.0
3  0.96499997E 00  0.0                0.0                0.0
4  0.75000000E 02  0.0                0.0                0.0
5  0.75000000E 02  0.0                0.0                0.0

J          B1(J)              B2(J)                B3(J)              B4(J)
1  0.0             0.29999996E-03  0.0                0.0
2  0.27999997E 00  0.0                0.0                0.0
3  0.12500000E 00  0.0                0.0                0.0
4  0.33999998E-01  0.0                0.0                0.0
5  0.33999998E-01  0.0                0.0                0.0

         RHO(I), I=1,   5
   0.10800000E 03     0.15000000E 03     0.33400000E 03     0.11920000E 04     0.11920000E 04

      EL(I), I=1,   5
   0.12500000E 00     0.99999979E-02     0.49999997E-01     0.25000000E 00     0.25000000E 00

     TEMP(I), I=1,   5
   0.54000000E 03     0.54000000E 03     0.54000000E 03     0.54000000E 03     0.54000000E 03

I     TIME      ALT      VINF    AOFA   HINSD   TINSD   QCCNS  QI*3600              TEMPW       VAPRES
1      0.0     200.0     10.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
2     22.0    4350.0    410.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
3     40.0   15500.0    915.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
4     54.0   19540.0   1410.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
5     60.0   37000.0   1690.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
6     67.0   47000.0   1850.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
7     70.0   51570.0   2000.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
8     80.0   69210.0   2630.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
9     96.0  104800.0   4000.0    0.0     0.0     0.0     1.0    0.0                 0.0         0.0
```

10	112.0	151300.0	5850.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
11	118.0	172125.0	7550.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
12	124.0	195210.0	8690.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
13	130.0	220540.0	9350.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
14	136.0	245180.0	9750.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
15	140.0	261700.0	9970.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
16	144.0	277990.0	10110.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
17	148.0	295000.0	10270.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

QSOLAR
0.0

END OF DATA PRINTOUT NDBUG= 0

```
COLD WALL INPUTS - AS PER ENTROPY SWALLOWING
NO. OF INPUTS = INPCW =    17
      QINT            HEDGE            PEDGE            UEDGE            HRECOV           TIME
0.39999964E-01   0.12300000E 03   0.99499995E 00   0.10000000E 02   0.12300000E 03   0.0
0.27999997E 00   0.12300000E 03   0.83999997E 00   0.40000000E 03   0.12300000E 03   0.22000000E 02
0.10000000E 01   0.12000000E 03   0.54999995E 00   0.90000000E 03   0.12600000E 03   0.40000000E 02
0.15900000E 02   0.13000000E 03   0.34999996E 00   0.13000000E 04   0.14000000E 03   0.54000000E 02
0.15599999E 02   0.16500000E 03   0.25000000E 00   0.15500000E 04   0.14600000E 03   0.60000000E 02
0.11360000E 02   0.16500000E 03   0.14999998E 00   0.17750000E 04   0.16000000E 03   0.67000000E 02
0.64199991E 01   0.16500000E 03   0.13000000E 00   0.18000000E 04   0.16700000E 03   0.70000000E 02
0.35699993E 01   0.17000000E 03   0.61999999E-01   0.24000000E 04   0.22100000E 03   0.80000000E 02
0.28799992E 01   0.19000000E 03   0.99999979E-02   0.36000000E 04   0.40000000E 03   0.96000000E 02
0.34299994E 01   0.2[illegible]0000E 03   0.33000000E-02   0.55000000E 04   0.70000000E 03   0.11200000E 03
0.41199999E 01   0.[illegible]0000E 03   0.13100000E-02   0.71490000E 04   0.10930000E 04   0.11800000E 03
0.36799994E 01   0.24000000E 03   0.65999990E-03   0.83200000E 04   0.13930000E 04   0.12400000E 03
0.25699997E 01   0.19900000E 03   0.26499992E-03   0.90840000E 04   0.15740000E 04   0.13000000E 03
0.15699997E 01   0.14900000E 03   0.94699993E-04   0.95930000E 04   0.16830000E 04   0.13600000E 03
0.10799999E 01   0.15300000E 03   0.42299987E-04   0.98020000E 04   0.17530000E 04   0.14000000E 03
0.71099997E 00   0.14000000E 03   0.18599996E-04   0.99770000E 04   0.18000000E 04   0.14400000E 03
0.37500000E 00   0.12700000E 03   0.76899996E-05   0.10000000E 05   0.19070000E 04   0.14800000E 03
INITIAL CONDITIONS
                                                                                         HSUBI
      TIME              ALT              VINF              ACFA           QSUBI
 0.0               0.0             0.27954150E 03    0.0               0.0             0.0
                                                                                         RECI
      QNET              REYN            DELTA             WABDCT          WAB
 0.0               0.0             0.49999997E-01    0.0               0.0             0.0
                                                                                         TABL
      WALLI             QSAV            QINSP             TRANR           HOABL
 0.12628716E 03    0.0             0.0               0.0               0.0             0.10000000E 05
                                                                                         QNSUM
      PSHK              DYNP            VSHK              SHKI            REFI
 0.18969067E 04    0.34EC4962E 02   0.0              0.0               0.0             0.0
                                                                                         WAB/SEC
      SHEAR             PS              PSTAG             EMIS            CCONS
 0.0               0.89645875E 00   0.93653667E 00   0.84999996E 00    0.10[illegible]0E 01    0.0

 1ST SEGMENT PRESENT=NSEG2 =   1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
     540.0        540.0         540.0    .    540.0        540.0
INITIAL CONDITIONS

 TRANSITION (COLD WALL) HAS OCCURRED
FROM ***  TO    2
                                                                                         HSUBI
      TIME              ALT              VINF              ACFA           QSUBI
 0.49999997E-01    0.20943181E 03   0.10999090E 02   0.0               0.65637946E-01  0.0
```

```
      QNET             REYN            DELTA           WABDOT          WAB             RECI
 0.30704618E-01   0.11796969E 05   0.49999997E-01   0.0             0.0             0.12300215E 03
      WALLI            QSAV            QINSD           TRANR           HOABL           TABL
 0.12838576E 03   0.0              0.0              0.0             0.99461621E 03  0.10000000E 05
      PSHK             DYNP            VSHK            SHKI            REFI            QNSUM
 0.21054697E 04   0.14025348E 00   0.10886363E 02   0.12300000E 03  0.12569334E 03  0.76761539E-03
      SHEAR            PS              PSTAG           EMIS            QCONS           WAB/SEC
-0.33182148E-02   0.99502349E 00   0.99261475E 00   0.84999996E 00  0.10000000E 01  0.0

 1ST SEGMENT PRESENT=NSEG2 =   1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
    540.0        540.0        540.0        540.0        540.0
STABILITY  CHECK,  DEL, CCN, CAY, WRI, DELTA
     0.05695309          0.25000000          0.02499999       200.00000000        0.02500000     -1

      TIME             ALT             VINF            ACFA            QSUBI           HSUBI
 0.49749384E 01   0.11294541E 04   0.10045340E 03   0.0             0.11288923E 00  0.0
      QNET             REYN            DELTA           WABDOT          WAB             RECI
 0.77648222E-01   0.10269356E 06   0.24999999E-01   0.0             0.0             0.12317575E 03
      WALLI            QSAV            QINSD           TRANR           HOABL           TABL
 0.12867006E 03   0.0              0.0              0.0             0.99450562E 03  0.10000000E 05
      PSHK             DYNP            VSHK            SHKI            REFI            QNSUM
 0.20320203E 04   0.11571953E 02   0.98192062E 02   0.12300000E 03  0.12587367E 03  0.26758307E 00
      SHEAR            PS              PSTAG           EMIS            QCONS           WAB/SEC
-0.50437782E-01   0.96031201E 00   0.96510357E 00   0.84999996E 00  0.10000000E 01  0.0

 1ST SEGMENT PRESENT=NSEG2 =   1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
    541.2        541.0        540.5        540.1        540.0

      TIME             ALT             VINF            ACFA            QSUBI           HSUBI
 0.99748621E 01   0.20816216E 04   0.19136111E 03   0.0             0.15560943E 00  0.0
      QNET             REYN            DELTA           WABDOT          WAB             RECI
 0.11990416E 00   0.18822169E 06   0.24999999E-01   0.0             0.0             0.12363628E 03
      WALLI            QSAV            QINSD           TRANR           HOABL           TABL
 0.12909567E 03   0.0              0.0              0.0             0.99454053E 03  0.10000000E 05
      PSHK             DYNP            VSHK            SHKI            REFI            QNSUM
 0.19574524E 04   0.40838104E 02   0.18682709E 03   0.12300000E 03  0.12619781E 03  0.76145631E 00
      SHEAR            PS              PSTAG           EMIS            QCONS           WAB/SEC
-0.13312852E 00   0.92507201E 00   0.94642693E 00   0.84999996E 00  0.10000000E 01  0.0

 1ST SEGMENT PRESENT=NSEG2 =   1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
    543.0        542.7        541.6        540.4        540.3

      TIME             ALT             VINF            ACFA            QSUBI           HSUBI
 0.14824789E 02   0.29964939E 04   0.27954150E 03   0.0             0.20843077E 00  0.0
      QNET             REYN            DELTA           WABDOT          WAB             RECI
 0.17238791E 00   0.26468369E 06   0.24999999E-01   0.0             0.0             0.12435663E 03
```

```
      WALLI            QSAV             QINSD            TRANR            HOABL            TABL
 0.12967349E 03    0.0              0.0              0.0              0.99468311E 03   0.10000000E 05
      PSHK             DYNP             VSHK             SHKI             REFI             QNSUM
 0.18851216E 04    0.84804562E 02   0.27280298E 03   0.12300000E 03   0.12663519E 03   0.14695253E 01
      SHEAR            PS               PSTAG            EMIS             QCONS            WAB/SEC
-0.26735902E 00    0.89088917E 00   0.93653667E 00   0.84999996E 00   0.10000000E 01   0.0

1 1 SEGMENT PRESENT=NSEG2 =   1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
     545.4       545.0       543.2       541.0       540.9
```

(NT = 592)

6... ONE DIMENSIONAL, FIVE SEGMENTS, NON-AEROHEATING, Q CORRECTION

```
 INPUT CARDS READ, DATA SET      3
&BCD6CS QCONS=5*.495,
&END
KARC =       2,    TO GO =     24

END OF INPUT.                                                                                   0
```

```
SAMPLE ONE DIMENSIONAL PROBLEM, - WITH COLD WALL INPUTS, 5 SEG.
IDENTIFICATION = &BDC60S
KASE    2
  CONE
INTEGERS
    ICONF      ICHK     IRANK     INPCW    ISTETS      KASE     KFUNT     NBCOL     NDBUG      NDIM
        3         0         0        17         0         2         3         0         0         1
   NCPSHR    NOPSTG      NOSB     NPGAS       NPQ       NPU       NQI      NSEG     NTRAJ    NOCHEK
        0         0         0         0         0         0         0         5        17         0
        0
```

```
SAMPLE ONE DIMENSIONAL PROBLEM, - WITH COLD WALL INPUTS, 5 SEG.

   TRANR             TRANLT          ANGLE              SWEEP              XDIST              CONPR
0.0               0.47000000E 05  0.80000000E 01  0.0                 0.24299997E 00  0.0

     DELTA              WRITE              STOP
   0.49999997E-01     0.50000000E 01     0.14799999E 02

     TABL               HV                 ETAL               ETAT               EMIS
   0.10000000E 05     0.10000000E 04     0.10000000E 01     0.10000000E 01     0.84999996E 00

          A1(J)            A2(J)              A3(J)              A4(J)
1  0.60000000E 02  0.0            0.0                0.0
2  0.25000000E 00  0.0            0.0                0.0
3  0.96499997E 00  0.0            0.0                0.0
4  0.75000000E 02  0.0            0.0                0.0
5  0.75000000E 02  0.0            0.0                0.0

J         B1(J)            B2(J)              B3(J)              B4(J)
1  0.0             0.29999996E-03 0.0                0.0
2  0.27999997E 00  0.0            0.0                0.0
3  0.12500000E 00  0.0            0.0                0.0
4  0.33999998E-01  0.0            0.0                0.0
5  0.33999998E-01  0.0            0.0                0.0

      RHO(I), I=1,  5
   0.10800000E 03     0.15000000E 03     0.33400000E 03     0.11920000E 04     0.11920000E 04

      EL(I), I=1,  5
   0.12500000E 00     0.99999979E-02     0.49999997E-01     0.25000000E 00     0.25000000E 00

      TEMP(I), I=1,  5
   0.54000000E 03     0.54000000E 03     0.54000000E 03     0.54000000E 03     0.54000000E 03

I   TIME     ALT      VINF   AOFA  HINSD  TINSD  QCONS  QI*3600        TEMPW    VAPRES
1    0.0    200.0     10.0   0.0    0.0    0.0    0.5    0.0           0.0      0.0
2   22.0   4350.0    410.0   0.0    0.0    0.0    0.5    0.0           0.0      0.0
3   40.0  15500.0    915.0   0.0    0.0    0.0    0.5    0.0           0.0      0.0
4   54.0  19540.0   1410.0   0.0    0.0    0.0    0.5    0.0           0.0      0.0
5   60.0  37000.0   1690.0   0.0    0.0    0.0    0.5    0.0           0.0      0.0
6   67.0  47000.0   1650.0   0.0    0.0    0.0    1.0    0.0           0.0      0.0
7   70.0  51570.0   2000.0   0.0    0.0    0.0    1.0    0.0           0.0      0.0
8   80.0  69210.0   2630.0   0.0    0.0    0.0    1.0    0.0           0.0      0.0
9   96.0 104800.0   4000.0   0.0    0.0    0.0    1.0    0.0           0.0      0.0
```

10	112.0	151300.0	5850.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
11	118.0	172125.0	7550.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
12	124.0	195210.0	8690.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
13	130.0	220540.0	9350.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
14	136.0	245180.0	9750.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
15	140.0	261700.0	9970.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
16	144.0	277990.0	10110.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
17	148.0	295000.0	10270.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

QSOLAR
0.0

END OF CATA PRINTOUT

NDBUG= 0

CCLD WALL INPUTS - AS PER ENTROPY SHALLOWING
NO. OF INPUTS = INPCW = 17

QINT	HEDGE	PEDGE	UEDGE	HRECOV	TIME
0.99999964E-01	0.12300000E 03	0.99499995E 00	0.10000000E 02	0.12300000E 03	0.0
0.27999997E 00	0.12300000E 03	0.83999997E 00	0.40000000E 03	0.12300000E 03	0.22000000E 02
0.10000000E 01	0.12000000E 03	0.54999995E 00	0.90000000E 03	0.12600000E 03	0.40000000E 02
0.15900000E 02	0.13000000E 03	0.34999996E 00	0.13000000E 04	0.14000000E 03	0.54000000E 02
0.15599999E 37	0.16500000E 03	0.25000000E 00	0.15500000E 04	0.14600000E 03	0.69000000E 02
0.11360000E 02	0.16500000E 03	0.14999998E 00	0.17750000E 04	0.16000000E 03	0.67000000E 02
0.64199991E 01	0.16500000E 03	0.13000000E 00	0.18000000E 04	0.16700000E 03	0.70000000E 02
0.35699993E 01	0.17000000E 03	0.61999999E-01	0.24000000E 04	0.22100000E 03	0.80000000E 02
0.28799992E 01	0.19000000E 03	0.99999979E-02	0.36000000E 04	0.43000000E 03	0.96000000E 02
0.34299994E 01	0.21000000E 03	0.33000000E-02	0.55000000E 04	0.72000000E 03	0.11200000E 03
0.41199999E 01	0.24200000E 03	0.13100000E-02	0.71490000E 04	0.10930000E 04	0.11800000E 03
0.36799994E 01	0.24000000E 03	0.65999990E-03	0.83200000E 04	0.13930000E 04	0.12400000E 03
0.25699997E 01	0.19900000E 03	0.26499992E-03	0.90840000E 04	0.15740000E 04	0.13000000E 03
0.15699997E 01	0.14900000E 03	0.94699993E-04	0.95930000E 04	0.16830000E 04	0.13600000E 03
0.10799999E 01	0.15300000E 03	0.42299997E-04	0.98020000E 04	0.17530000E 04	0.14000000E 03
0.71099997E 00	0.14000000E 03	0.18599996E-04	0.99770000E 04	0.18000000E 04	0.14400000E 03
0.37500000E 00	0.12700000E 03	0.76899996E-05	0.10000000E 05	0.19700000E 04	0.14800000E 03

```
INITIAL CONDITIONS
        TIME              ALT               VINF              ACEA              QSUBI             HSUBI
 0.0               0.0              0.27954150E 03    0.0               0.0               0.0
        GNET              REYN              DELTA             WABDOT            WAB               RECI
 0.0               0.0              0.49999997E-01    0.0               0.0               0.0
        WALLI             QSAV              QINSD             TPANR             HOABL             TABL
 0.12967349E 03    0.0              0.0               0.0               0.99468311E 03    0.10000000E 05
        PSHK              DYNP              VSHK              SHKI              PFFI              QNSUM
 0.18851216E 04    0.84804942E 02   0.0               0.0               0.0               0.0
        SHEAR             PS                PSTAG             EMIS              QCONS             WAB/SEC
 0.0               0.89088917E 00   0.93653667E 00    0.84999996E 00    0.10000000E 01    0.0

1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I),  =NSEG2,NSEG FOLLOW.
     540.0        540.0        540.0        40.0        540.0
INITIAL CONDITIONS
        TIME              ALT               VINF              ACEA              QSUBI             HSUBI
 0.49999997E-01    0.20943181E 03   0.10909090E 02    0.0               0.65637946E-01    0.0
        GNET              REYN              DELTA             WABDOT            WAB               RECI
-0.24425506E-02    0.11796969E 05   0.49999997E-01    0.0               0.0               0.12300215E 03
        WALLI             QSAV              QINSD             TPANR             HOABL             TABL
 0.12338576E 03    0.0              0.0               0.0               0.99468311E 03    0.10000000E 05
```

```
     PSHK            DYNP            VSHK            SHKI            REFI            QNSUM
 0.21054697E 04  0.14025348E 00  0.10886363E 02  0.12300000E 03  0.12569334E 03  0.0
     SHEAR           PS              PSTAG           EMIS            QCONS           WAB/SEC
-0.16425161E-02  0.95502349E 00  0.99261475E 00  0.84999996E 00  0.49499995E 00  0.0

 1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
     540.0       540.0       540.0       540.0       540.0
STABILITY  CHECK,  DEL, CCN, CAY, WRI, DELTA
       0.05695309          0.25000000          0.02499999          200.00000000          0.02500000      -1

     TIME            ALT             VINF            ACFA            QSUBI           HSUBI
 0.49749384E 01  0.11384541E 04  0.10045340E 03  0.0             0.10804212E 00  0.0
     QNET            REYN            DELTA           WABDOT          WAB             RECI
 0.18491846E-01  0.10269356E 06  0.24999999E-01  0.0             0.0             0.12317575E 03
     WALLI           QSAV            QINSD           TRANR           HOABL           TABL
 0.12843733E 03  0.0             0.0             0.0             0.99473828E 03  0.10000000E 05
     PSHK            DYNP            VSHK            SHKI            REFI            QNSUM
 0.20320203E 04  0.11571953E 02  0.98192062E 02  0.12300000E 03  0.12575731E 03  0.39520875E-01
     SHEAR           PS              PSTAG           EMIS            QCONS           WAB/SEC
-0.24951588E-01  0.96031201E 00  0.96510357E 00  0.84999996E 00  0.49499995E 00  0.0

 1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
     540.2       540.2       540.1       540.0       540.0

     TIME            ALT             VINF            ACFA            QSUBI           HSUBI
 0.99748621E 01  0.20816216E 04  0.19136111E 03  0.0             0.14013332E 00  0.0
     QNET            REYN            DELTA           WABDOT          WAB             RECI
 0.342487[illegible]    01  0.18822169E 06  0.24999999E-01  0.0             0.0             0.12363628E 03
     WALLI           QSAV            QINSD           TRANR           HOABL           TABL
 0.12855637E 03  0.0             0.0             0.0             0.99507983E 03  0.10000000E 05
     PSHK            DYNP            VSHK            SHKI            REFI            QNSUM
 0.19574524E 04  0.40838104E 02  0.18682779E 03  0.12300000E 03  0.12591815E 03  0.17136908E 00
     SHEAR           PS              PSTAG           EMIS            QCONS           WAB/SEC
-0.65849602E-01  0.92507201E 00  0.94642693E 00  0.84999996E 00  0.49499995E 00  0.0

 1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
     540.7       540.6       540.4       540.1       540.1

     TIME            ALT             VINF            ACFA            QSUBI           HSUBI
 0.14824788E 02  0.29964939E 04  0.27954150E 03  0.0             0.17121571E 00  0.0
     QNET            REYN            DELTA           WABDOT          WAB             RECI
 0.49448051E-01  0.26468369E 06  0.24999999E-01  0.0             0.0             0.12435663E 03
     WALLI           QSAV            QINSD           TRANR           HOABL           TABL
 0.12872774E 03  0.0             0.0             0.0             0.99562866E 03  0.10000000E 05
     PSHK            DYNP            VSHK            SHKI            REFI            QNSUM
 0.18851216E 04  0.84804962E 02  0.27280299E 03  0.12300000E 03  0.12616231E 03  0.37432923E 00
```

SHEAR	PS	PSTAG	EMIS	QCONS	WAB/SEC
-).13223493E CC	C.89C88917E 00	0.93653667E 00	0.84999996E 00	0.49499995E 00	0.0

```
1ST SEGMENT PRESENT=NSEG2 =   1.   TEMP(I), I=NSEG2,NSEG FOLLOW.
    541.4        541.3        540.8        540.2        540.2
```

(NT = 5?2)

6.4 ONE DIMENSIONAL, SIX SEGMENTS, NON-AEROHEATING, NO Q CORRECTION

```
 INPUT CARDS READ, DATA SET      4
&3D)6CS  TITLE(15)=', SI','X SE','GMEN','TS  ',    KASE=3,
NSEG=6,
CCCNS=6*1.,
TEMP=6*540.,
A1=.14,60.,.25,.965,2*75.,
31=.26,.0,.28,.125,2*.034, B2=.0,.0003,4*.0,
RHO=136.,108.,150.,334.,2*1192.,
EL=.010,.125,.010,.050,2*.25,
EMIS=.15,HV=950.,ETAL=.43,ETAT=.17,TABL=1800.,
&END     NOTE - CHANGED EMIS, HV, TABL, ECT.
KARD =      10,    TO GO =     14

END OF INPUT.                                                    0
```

```
 SAMPLE ONE DIMENSIONAL PROBLEM, - WITH COLD WALL INPUTS, SIX SEGMENTS
IDENTIFICATION = &BD060S
KASE    3
  CONE
INTEGERS
    ICONF       ICHK      IHANK      INPCW     ISTETS       KASE      KFUNT      NBCOL      NDBUG       NDIM
        3          0          0         17          0          3          3          0          0          1
   NCPSHR     NOPSTG       NOSB      NPGAS        NPQ        NPU        NQI       NSEG      NTRAJ     NOCHEK
        0          0          0          0          0          0          0          6         17          0
        0
```

SAMPLE ONE DIMENSIONAL PROBLEM, - WITH COLD WALL INPUTS, SIX SEGMENTS

```
    TRANR            TRANLT          ANGLE               SWEEP            XDIST             CONPR
0.0              0.47000000E 05  0.80000000E 01  0.0                  0.24299997E 00  0.0

     DELTA              WRITE              STOP
    0.49999997E-01     0.50000000E 01     0.14799999E 02

     TABL               HV                 FTAL               ETAT               EMIS
    0.18000000E 04     0.95000000E 03     0.42999995E 00     0.16999996E 00     0.14999998E 00

J        A1(J)             A2(J)             A3(J)             A4(J)
1  0.13999999E 00  0.0             0.0             0.0
2  0.60000000E 02  0.0             0.0             0.0
3  0.25000000E 00  0.0             0.0             0.0
4  0.96499997E 00  0.0             0.0             0.0
5  0.75000000E 02  0.0             0.0             0.0
6  0.75000000E 02  0.0             0.0             0.0

J        B1(J)             B2(J)             B3(J)             B4(J)
1  0.25999999E 00  0.0             0.0             0.0
2  0.0             0.29999996E-03  0.0             0.0
3  0.27999997E 00  0.0             0.0             0.0
4  0.12500000E 00  0.0             0.0             0.0
5  0.33999998E-01  0.0             0.0             0.0
6  0.33999998E-01  0.0             0.0             0.0

        RHO(I), I=1,   6
    0.13600000E 03     0.10800000E 03     0.15000000E 03     0.33400000E 03     0.11920000E 04
    0.11920000E 04

        EL(I), I=1,   6
    0.99999979E-02     0.12500000E 00     0.99999979E-02     0.49999997E-01     0.25000000E 00
    0.25000000E 00

      TEMP(I), I=1,   6
    0.54000000E 03     0.54000000E 03     0.54000000E 03     0.54000000E 03     0.54000000E 03
    0.54000000E 03

I    TIME     ALT      VINF    AOFA   HINSD  TINSD  QCCNS  QI*3600        TEMPW     VAPRES
1     0.0    200.0     10.0    0.0    0.0    0.0    1.0    0.0             0.0       0.0
2    22.0   4350.0    410.0    0.0    0.0    0.0    1.0    0.0             0.0       0.0
3    40.0  15500.0    915.0    0.0    0.0    0.0    1.0    0.0             0.0       0.0
4    54.0  19540.0   1410.0    0.0    0.0    0.0    1.0    0.0             0.0       0.0
```

5	60.0	37000.0	1690.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
6	67.0	47000.0	1850.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
7	70.0	51570.0	2000.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
8	80.0	69210.0	2630.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
9	96.0	104800.0	4000.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
10	112.0	151300.0	5850.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
11	118.0	172125.0	7550.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
12	124.0	195210.0	8690.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
13	130.0	220540.0	9350.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
14	136.0	245180.0	9750.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
15	140.0	261700.0	9970.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
16	144.0	277990.0	10110.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
17	148.0	295000.0	10270.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

QSOLAR
0.0

END OF DATA PRINTOUT

NDBUG= 0

EDCB WALL INPUTS - AS PER ENTROPY SWALLOWING
NO. OF INPUTS = INPCW = 17

QINT	HEDGE	PEDGE	UEDGE	HRECOV	TIME
0.99999964E-01	0.12300000E C3	0.99499995E 00	0.10000000E 02	0.12300000E 03	0.0
0.27999997E 00	C.12300000E C3	0.83999997E 00	0.40000000E 03	0.12300000E 03	0.22000000E 02
0.10000000E 01	0.12000000E C3	0.54999995E 00	0.90000000E 03	0.12600000E 03	0.40000000E 02
0.15700000E 02	C.13000000E C3	0.34999996E 00	0.13000000E 04	0.14000000E 03	0.54000000E 02
0.15999999E 02	0.16500000E C3	0.25000000E 00	0.15500000E 04	0.14600000E 03	0.60000000E 02
0.11360000E 02	0.16500000E C3	0.14999998E 00	0.17750000E 04	0.16000000E 03	0.67000000E 02
0.64199991E 01	0.16500000E C3	0.13000000E 00	0.18000000E 04	0.16700000E 03	0.70000000E 02
0.39699993E 01	C.17000000E C3	0.61999999E-01	0.24000000E 04	0.22100000E 03	0.80000000E 02
0.28799992E 01	0.19000000E C3	0.99999979E-02	0.36000000E 04	0.40000000E 03	0.96000000E 02
0.14299994E 01	0.21000000E C3	0.33000000E-02	0.55000000E 04	0.70000000E 03	0.11200000E 03
0.41199999E 01	0.24200000E C3	0.13100000E-02	0.71490000E 04	0.10930000E C4	C.11800000E 03
0.36799994E 01	0.24000000E C3	0.65999990E-03	0.83200000E 04	0.13930000E 04	0.12400000E 03
0.25699997E 01	C.19900000E C3	0.26499992E-03	0.90840000E 04	0.15740000E 04	0.13000000E 03
0.15699997E 01	C.14900000E C3	C.94699993E-04	0.95930000E 04	C.16830000E 04	0.13600000E 03
0.10799999E 01	0.15300000E C3	C.42299987E-04	0.98020000E 04	0.17530000E 04	0.14000000E 03
0.71099997E 00	0.14000000E C3	0.18599996E-04	0.99770000E 04	C.18000000E 04	0.14400000E 03
0.17500000E 00	C.12700000E C3	C.76899996E-05	0.10000000E 05	0.19070000E 04	C.14800000E 03

INITIAL CONDITIONS

TIME	ALT	VINF	ACFA	QSUBI	HSUBI	
0.0	0.0	0.27954150E 03	0.0	0.0	0.0	
QNET	REYN	DELTA	WABDOT	WAB	RECI	
0.0	0.0	0.49999997E-01	0.0	0.0	0.0	
WALLT	QSAV	QINSD	TRANR	HOABL	TABL	
0.12872774E 03	0.0	0.0	0.0	0.99562866E 03	0.18000000E 04	
PSHK	DYNP	VSHK	SHKI	REFI	QNSUM	
0.18851216E 04	0.84804562E 02	0.0	0.0	0.0	0.0	
SHEAR	PS	PSTAG	EMIS	QCONS	WAB/SEC	
0.0	0.89C88917E 00	0.93653667E 00	0.14999998E 00	0.49499995E 00	0.0	

1ST SEGMENT PRESENT=NSEG2 = 1. TEMP(I), I=NSEG2,NSEG FOLLOW.
540.0 540.0 540.0 540.0 540.0
540.0

INITIAL CONDITIONS

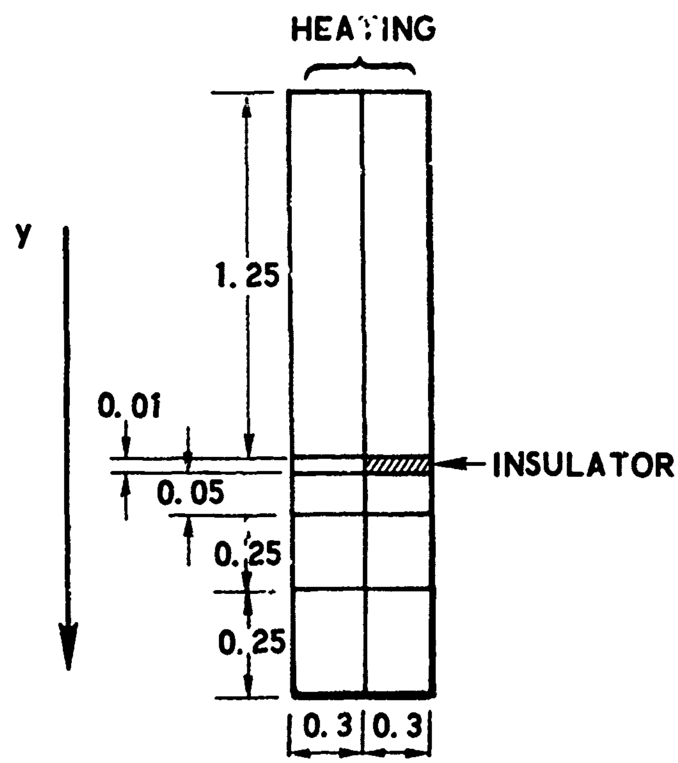
TIME	ALT	VINF	AOFA	QSUBI	HSUBI
0.49999997E-01	0.20943181E 03	0.10909090E 02	0.0	0.65637946E-01	0.0
QNET	REYN	DELTA	WABDOT	WAB	RECI
[illegible]E-01	0.11796969E 05	0.49999997E-01	0.0	0.0	0.12300215E 03
WALLT	QSAV	QINSD	TRANR	HOABL	TABL

```
0.12838576E 03    0.0               0.0               0.0               0.94908472E 03    0.18000000E 04
      PSHK              CYNP              VSHK              SHKI              REFI              QNSUM
0.21054697E 04   -0.14025248E 00    0.10836363E 02    0.12300000E 03    0.12569334E 03    0.14868309E-02
      SHEAR             PS                PSTAG             EMIS              QCONS             WAB/SEC
-0.33142148E-02   0.95592345E 00    0.99261475E 00    0.14999958E 00    0.10000000E 01    0.0

1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
   540.1      540.0      540.0      540.0      540.0
   540.0
STABILITY  CHECK,  DFL, CCN, CAY, WFI, DELTA
      0.05695339        0.25000000        0.12499999        200.00000000        0.02500000    -1

      TIME              ALT               VINF              ACFA              QSUBI             HSUBI
0.49749384E 01    0.11384541E 04    0.13245340E 03    0.0               0.12011385E 00    0.0
      QNET              REYN              DELTA             WABDCT            WAB               RECI
0.11382318E 00    0.10269356E 06    0.24999999E-01    0.0               0.0               0.12317575E 03
      WALLI             QSAV              QINSD             TRANR             HOABL             TABL
0.12901635E 03    0.0               0.0               0.0               0.94900708E 03    0.18000000E 04
      PSHK              CYNP              VSHK              SHKI              REFI              QNSUM
0.20320203E 04   -0.11571953E 02    0.98192062E 02    0.12300000E 03    0.12604684E 03    0.42823595E 00
      SHEAR             PS                PSTAG             EMIS              QCONS             WAB/SEC
-0.50493465E-01   0.96031201E 00    0.96519357E 00    0.14999958E 00    0.10000000E 01    0.0

1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
   542.6      541.5      541.3      540.6      540.1
   540.1

      TIME              ALT               VINF              ACFA              QSUBI             HSUBI
0.94748621E 01    0.20816216E 04    0.19136111E 03    0.0               0.17183381E 00    0.0
      QNET              REYN              DELTA             WABDCT            WAB               RECI
0.16542292E 00    0.18822169E 06    0.24999999E-01    0.0               0.0               0.12363628E 03
      WALLI             QSAV              QINSD             TRANR             HOABL             TABL
0.12966314E 03    0.0               0.0               0.0               0.94897583E 03    0.18000000E 04
      PSHK              CYNP              VSHK              SHKI              REFI              QNSUM
0.19574524E 04   -0.40838104E 02    0.18682709E 03    0.12300000E 03    0.12647003E 03    0.11263523E 01
      SHEAR             PS                PSTAG             EMIS              QCONS             WAB/SEC
-0.13223343E 00   0.92507201E 00    0.94642693E 00    0.14999958E 00    0.10000000E 01    0.0

1ST SEGMENT PRESENT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
   545.3      543.7      543.4      542.0      540.5
   540.4

      TIME              ALT               VINF              ACFA              QSUBI             HSUBI
0.14824788E 02    0.29964939E 04    0.27954151E 03    0.0               0.24147457E 00    0.0
      QNET              REYN              DELTA             WABDOT            WAB               RECI
0.23439529E 00    0.26468369E 06    0.24999999E-01    0.0               0.0               0.12435663E 03
```

Reproduced from best available copy.

```
      WALLI              QSAV              GINSD              TRANR              HOABL              TABL
 0.13C51186E 03    0.0               0.0               0.0               C.94895361E 03    0.18000000F 04
      PSHK               CYNP              VSHK               SHKI               REFI               QNSUM
 0.18851216E 04    C.84204962E 02    0.27280298E 03    0.12300000E 03    0.12705437E 03    0.20971088E 01
      SHEAR              PS                PSTAG              EMIS               QCONS              WAB/SEC
-0.26755697E 00    C.89C88917E CC    0.93653667E 00    0.14999998E 00    0.10000000E 01    0.0

1ST SEGMENT PRESFNT=NSEG2 =  1.  TEMP(I), I=NSEG2,NSEG FOLLOW.
     548.9       546.6       546.1       544.0       541.2
     541.1
```

6.5 TWO DIMENSIONAL, FIVE SEGMENTS x TWO SLABS, NON-AEROHEATING



71 2613502

Figure 6-1. Diagram of Two-Dimensional Problem (Section 6.5)

```
 INPUT CARDS READ, DATA SET      5
&9C)60S  TITLE=' TWO-DIMENSIONAL PRUBLEM, BASED ON KASE 3 (1-D)',
KASE=23, NDIM=2,
WRITE=.C2,
START= 68.7,
&ENC
KARC =        5,    TO GC =      9

ENC OF INPUT.
```

```
 TWO-DIMENSIONAL PROBLEM, BASED ON KASE 3 (1-D)
IDENTIFICATION = &BD060S
KASE  23
  CONE
INTEGERS
    ICONF       ICHK       IHANK       INPCW      ISTETS       KASE      KFUNT      NBCOL      NDBUG       NDIM
        3          0           0          17           0         23          3          0          0          2
   NCPSHR     NCPSTG        NOSB       NPGAS         NPQ        NPU        NQI       NSEG      NTRAJ     NOCHEK
        0          0           0           0           0          0          ,          6         17          0
        0


TWO-DIMENSIONAL OPTION BEING USED


NARCW NBCOL NDIM
    5     2     2
THICKNESS  Y,AA
 0.12500000E 00  0.99999979E-02  0.49999997E-01  0.25000000E 00  0.25000000E 00
WICTH      X,BB
 0.30799997E 00  0.30799997E 00
COND. CONST.    ,CC
0.10000E 01 0.10000E 01 0.0         0.0        0.0        0.0        0.0        0.0        0.0
0.10000E 01 0.0         0.0         0.0        0.0        0.0        0.0        0.0        0.0
0.10000E 01 0.10000E 01 0.0         0.0        0.0        0.0        0.0        0.0        0.0
0.10000E 01 0.10000E 01 0.0         0.0        0.0        0.0        0.0        0.0        0.0
0.10000E 01 0.10000E 01 0.0         0.0        0.0        0.0        0.0        0.0        0.0
TEMPERATURE (R),T2
0.65700E 03 0.66500E 03 0.0         0.0        0.0        0.0        0.0        0.0        0.0
0.63200E 03 0.0         0.0         0.0        0.0        0.0        0.0        0.0        0.0
0.57700E 03 0.54400E 03 0.0         0.0        0.0        0.0        0.0        0.0        0.0
0.54600E 03 0.54400E 03 0.0         0.0        0.0        0.0        0.0        0.0        0.0
0.54500E 03 0.54400E 03 0.0         0.0        0.0        0.0        0.0        0.0        0.0
```

```
TWO-DIMENSIONAL PROBLEM, BASED ON KASE 3 (1-D)

    TRANR            TRANLT          ANGLE             SWEEP            XDIST            CONPR
0.0               0.47000000E 05  0.80000000E 01  0.0               0.24299997E 00  0.0

      DELTA            WRITE            STOP
    0.49999997E-01   0.20000000E-01   0.14799999E 02

      TABL             HV               ETAL             ETAT             EMIS
    0.18000000E 04   0.95000000E 03   0.42999995E 00   0.16999996E 00   0.14999998E 00

J          A1(J)           A2(J)            A3(J)            A4(J)
1   0.13999999E 00  0.0             0.0             0.0
2   0.60000000E 02  0.0             0.0             0.0
3   0.25000000E 00  0.0             0.0             0.0
4   0.96499997E 00  0.0             0.0             0.0
5   0.75000000E 02  0.0             0.0             0.0

J          B1(J)           B2(J)            B3(J)            B4(J)
1   0.25999999E 00  0.0             0.0             0.0
2   0.0             0.24999996E-03  0.0             0.0
3   0.27999997E 00  0.0             0.0             0.0
4   0.12500000E 00  0.0             0.0             0.0
5   0.33999998E-01  0.0             0.0             0.0

          RHO(I), I=1,  5
    0.13600000E 03   0.10800000E 03   0.15000000E 03   0.33400000E 03   0.11920000E 04

 5 VALUES OF AA
    0.12500000E 00   0.99999979E-02   0.49999997E-01   0.25000000E 00   0.25000000E 00

 2 VALUES OF BB
    0.30799997E 00   0.30799997E 00

I , J          CC(I,J)          T2(I,J)
1   1       0.10000000E 01   0.65700000E 03
1   2       0.10000000E 01   0.66500000E 03
2   1       0.10000000E 01   0.63200000E 03
2   2       0.0              0.0
3   1       0.10000000E 01   0.57700000E 03
3   2       0.10000000E 01   0.54400000E 03
4   1       0.10000000E 01   0.54600000E 03
4   2       0.10000000E 01   0.54400000E 03
5   1       0.10000000E 01   0.54500000E 03
5   2       0.10000000E 01   0.54400000E 03
```

Reproduced from best available copy.

I	TIME	ALT	VINF	AOFA	HINSD	TINSD	QCONS	QI*3600	TEMPW	VAPRES
1	0.0	200.0	10.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
2	22.0	4350.0	410.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
3	40.0	15500.0	915.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
4	54.0	19540.0	1410.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
5	60.0	37000.0	1690.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
6	67.0	47000.0	1850.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
7	70.0	51570.0	2000.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
8	80.0	69210.0	2630.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
9	96.0	104800.0	4000.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
10	112.0	151300.0	5850.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
11	118.0	172125.0	7550.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
12	124.0	195210.0	8690.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
13	130.0	220540.0	9350.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
14	136.0	245180.0	9750.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
15	140.0	261700.0	9970.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
16	144.0	277990.0	10110.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
17	148.0	295000.0	10270.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

QSOLAR
0.0

END OF DATA PRINTOUT NDBUG= 0

COLD WALL INPUTS - AS PER ENTROPY SWALLOWING

NO. OF INPUTS = INPCH = 17

QINT	HEDGE	PEDGE	UEDGE	HRECOV	TIME
0.99999964E-01	0.12300000E 03	0.99499995E 00	0.10000000E 02	0.12300000E 03	0.0
0.27599997E 00	0.12300000E 03	0.83999997E 00	0.40000000E 03	0.12300000E 03	0.22000000E 02
0.10000000E 01	0.12000000E 03	0.54999995E 00	0.90000000E 03	0.12600000E 03	0.40000000E 02
0.15900000E 02	0.13000000E 03	0.34999996E 00	0.13000000E 04	0.14000000E 03	0.54000000E 02
0.15599999E 02	0.16500000E 03	0.25000000E 00	0.15500000E 04	0.14600000E 03	0.60000000E 02
0.11360000E 02	0.16500000E 03	0.14999998E 00	0.17750000E 04	0.16000000E 03	0.67000000E 02
0.64199991E 01	0.16500000E 03	0.13000000E 00	0.18000000E 04	0.16700000E 03	0.70000000E 02
0.35699993E 01	0.17000000E 03	0.61999999E-01	0.24000000E 04	0.22100000E 03	0.80000000E 02
0.28799992E 01	0.19000000E 03	0.99999979E-02	0.36000000E 04	0.40000000E 03	0.95000000E 02
0.34299994E 01	0.21000000E 03	0.33000000E-02	0.55000000E 04	0.70000000E 03	0.11[illegible]E 03
0.41199999E 01	0.24200000E 03	0.13100000E-02	0.71490000E 04	0.10930000E 04	0.1[illegible]0000E 03
0.36799994E 01	0.24000000E 03	0.65999990E-03	0.83200000E 04	0.13930000E 04	0.12[illegible]00000E 03
0.25699997E 01	0.19900000E 03	0.26499992E-03	0.90840000E 04	0.15740000E 04	0.13000000E 03
0.15699997E 01	0.14900000E 03	0.94699993E-04	0.95930000E 04	0.16830000E 04	0.13600000E 03
0.10799999E 01	0.15300000E 03	0.42299987E-04	0.98020000E 04	0.17530000E 04	0.14000000E 03
0.71099997E 00	0.14000000E 03	0.18599996E-04	0.99770000E 04	0.18000000E 04	0.14400000E 03
0.37500000E 00	0.12700000E 03	0.76899996E-05	0.10000000E 05	0.19070000E 04	0.14800000E 03

INITIAL CONDITIONS

TIME	ALT	VINF	ACFA	QSUBI	HSUBI
0.68699997E 02	0.68699997E 02	0.27954150E 03	0.0	0.0	0.0
GNET	REYN	DELTA	WABCCT	WAB	RECI
0.0	0.0	0.49999997E-01	0.0	0.0	0.0
WALLI	QSAV	QINSD	TRANR	HOABL	TABL
0.13051186E 03	0.0	0.0	0.0	0.94895361E 03	0.18000000E 04
PSHK	CYNP	VSHK	SHKI	REFI	GNSUM
0.18851216E 04	0.84804562E 02	0.0	0.0	0.0	0.0
SHEAR	PS	PSTAG	EMIS	QCONS	WAB/SEC
0.0	0.89088917E 00	0.93653667E 00	0.14999998E 00	0.10000000E 01	0.0

TEMPERATURES FOR 2D FOLLOW

657.0	665.0
632.0	0.0
577.0	544.0
546.0	544.0
545.0	544.0

INITIAL CONDITIONS

TRANSITION (COLD WALL) HAS OCCURRED
FROM 2 TO 1

```
       TIME                 ALT                  VINF                 ACFA                 QSUBI                HSUBI
0.68749985E 02      0.49665809E 05      0.19374990E 04      0.0                  0.59073496E 01      0.0
       QNET                 REYN                 DELTA                WABCCT               WAB                  RECI
0.58938408E 01      0.0                 0.49999997E-01      0.0                  0.0                 0.22077075E 03
       WALLI                QSAV                 QINSD                TRAAR                HOABL                TABL
0.15661850E 03      0.0                 0.0                 0.0                  0.94895361E 03      0.18000000E 04
       PSHK                 DYNP                 VSHK                 SHKI                 REFI                 QNSUM
0.29282397E 03      0.69251538E 03      0.17895830E 04      0.16500000E 03       0.17307881E 03      0.14731002E 00
       SHEAR                PS                   PSTAG                EMIS                 QCONS                WAB/SEC
0.41197662E 01      0.13838559E 00      0.65960044E 00      0.14999998E 00       0.10000000E 01      0.0

TEMPERATURES FOR 2D FOLLOW
   657.8      665.8
   627.2        0.0
   577.4      544.0
   546.0      544.0
   545.0      544.0
```

(NT = 1)

SECTION VII

RECOMMENDATIONS

7.1 PROGRAM PHYSICAL MODIFICATIONS

During the program checkout process it became evident that with a variable degree of effort, the physical modeling within ASAH could be expanded to enhance the flexibility and performance of the computer code. Toward this end, the following list of recommended program changes was compiled. Several of these recommendations have already been implemented within ASAH and are so indicated.

7.1.1 Nose Blunting Option

The nose blunting option has been expanded to include both laminar and turbulent considerations at either the sonic or tangency points. The computations are still constrained by assumed sphere/cone geometry. These modifications have been completed. However, the ultimate solution within this option would eliminate the spherical geometry restriction that currently plagues this subroutine. It is recommended that the nose blunting option be further expanded to include calculations at many points along the nose configuration and to utilize polynomial curve fitting techniques to the ablated nose surface.

7.1.2 Cylindrical Coordinates

It is recommended that a cylindrical coordinate system option be included in the conduction subroutine (ONED) to complement the present rectangular coordinate system. The cylindrical coordinate system would be particularly useful in the nosetip region of a reentry vehicle.

7.1.3 Surface Temperature

The present ASAH computes temperatures at the midpoints of the nodal network. Since incipient ablation, reference flow field properties and the heat balance at the surface are dependent upon surface temperature, it is advisable to provide as accurate an estimate for surface temperature as possible. Therefore, it is recommended that

some extrapolation or analytical technique be incorporated into this program to allow determination of the surface temperature from the nodal midpoint.

7.1.4 Angle of Attack

At present ASAH accounts for angle of attack effects by simply adding the input angle of attack to the body angle to determine an effective flow incidence angle. Although this additive technique is adequate for very small angles of attack, it deteriorates rapidly as angle of attack increases. It is recommended that a more sophisticated angle of attack routine be included.

7.1.5 Ablation Temperature

At present, ASAH performs ablation computations holding surface temperature at a prescribed limiting value. It is recommended that the program be changed to accept a variable wall temperature as a boundary condition.

7.1.6 Mixed Flow Boundary Layer Growth

Heat transfer computations are performed assuming an all laminar or all turbulent boundary layer growth. Under actual flight conditions the boundary layer growth will be, at some point, determined by a partial run of laminar and turbulent flow conditions. The computer code should be modified to allow for this contingency by matching displacement or momentum thickness at the transition station.

7.1.7 Prandtl Number

The computer code performs calculations using a constant Prandtl number. Recent work by H. Jacobs of Aerospace has indicated that Prandtl number is a function of stream energy. Jacob's Prandtl number correlation should be added to ASAH.

7.1.8 Plotting Option

At present, the code does not have a plotting capability. Interpretation and assimilation of heat transfer results (heat transfer rate, mass loss rate, surface temperature, and backface temperature) could be improved if a plotting option were included in the computer code.

7.1.9 Insulation Thickness Selection

The usefulness of ASAH would be increased if the controls of the program were expanded to include a recycle feature that sized a heatshield based upon perhaps a backface temperature design criterion. With this option a heatshield thickness could be determined with a single computer simulation.

7.1.10 Wetted Distance Adjustment

For computations near the nosetip, wetted distance may significantly change as nosetip recession proceeds to grind away at the RV body. It is recommended that a change be made to ASAH to provide for variable wetted distance as a function of nosetip recession. This particular option would be most useful in the strong entropy swallowing regime near the RV nose.

REFERENCES

1. D. J. Smith, and L. Fisher, Analytical Methods and Equations for Use in an IBM 704 Program to Determine the Effects of Aerodynamic Heating by the Reference Enthalpy Method, Convair Report TC-15.

2. W. D. McCauley, "Similarity of Reentry Vehicle and Decoy Ballistic Parameters During Reentry," Proceedings 22nd AMRAC Meeting (Oct 1966).

3. Minzer, Champion and Pond, "ARDC Model Atmosphere - 1959."

4. Kemp and Riddell, "Heat Transfer to Satellite Vehicles," Jet Propulsion, Vol. 27, Number 12.

5. E. R. Eckert, Survey of Heat Transfer at High Speeds, WADC Technical Report 54-70.

6. Schadt and Livett, Aerodynamic Heating Using Real Properties of Air Behind Shock Waves, Applied Research Thermo Group Report No. ZR-658-024 (23 December 1958).

7. C. F. Hansen, Approximations for the Thermodynamic and Transport Properties of High Temperature Air, NACA TN 4150.

8. M. F. Romig, The Reference Temperature Method for Calculating Skin Friction and Heat Transfer in Compressible Flow, Convair Report A-ATLAS-151 (16 December 1954).

9. B. J. Griffith, and C. H. Lewis, "Laminar Heat Transfer to Spherically Blunted Cones at Hypersonic Conditions," AIAA Vol. 2, Number 3 (March 1964) pp. 439-444.

10. M. Sibulkin, "Heat Transfer Near the Stagnation Point of a Body Revolution," Journal of Aeronautical Sciences, Vol. 19, p. 570.

11. W. L. Hankey, E. H. Flinn, and R. D. Neuman, Design Procedures for Computing Aerodynamic Heating at Hypersonic Speed, Wright Air Development TR 59-610, AD 239 943 (June 1960).

12. K. F. Stetson, "Preliminary Cone Ablation Results," AIAA, Paper No. 64-433, June 1964.

13. G. H. Schadt, "Aerodynamic Heating of a Flat Plate at 90° to the Flow," Convair Report TG-184 (14 October 1959).

Preceding page blank

REFERENCES (Continued)

14. H. R. Jacobs, Engineering Approximation of the Effects of Blunting on Cones in Laminar and Turbulent Flow. SAMSO TR-68-258.

15. M. F. Romig, "On the Estimation of Compressible Heat Transfer for High Temperature Air," Convair Scientific Research Lab Memo (June 1968).

16. Eggers, Hansen, and Cunningham, Stagnation Point Heat Transfer to Blunt Shapes in Hypersonic Flow, Including the Effects of Yaw, NACA TN 4229.

17. IBM System/360 FORTRAN IV Language, IBM Form C28-6515-7 (October 1968), pp. 47-48.

18. L. Lees, "Laminar Heat Transfer over Blunt Bodies at Hypersonic Flight Speeds," Jet Propulsion, April 1956, Vol. 26, Nr. 4, pp. 259-269, 274.

BIBLIOGRAPHY

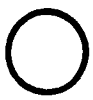
1. Georgiev and George, "Evaluation of Heatshield Ablation During Reentry of the Jupiter Nose Cone," Avco Research Note 81.

2. Sepulveda, J. A., and Love, C. C., "Analysis of the Smith-Fisher Aerodynamic Heating Program," Convair/Astronautics PDSA 34-61 (3 May 1961).

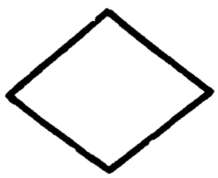
3. Taylor, R. G., and Love, C. C., "Use of the Smith-Fisher Program for Preliminary Analysis of Aerodynamic Heating," Astronautics Report PDAS 206-59 (21 December 1959).

APPENDIX A

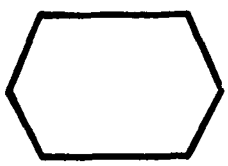
FLOW CHART NOMENCLATURE



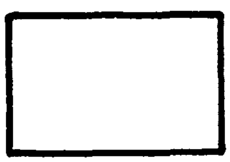
Statement Number



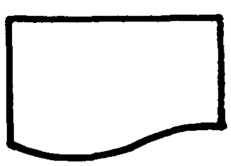
Decision



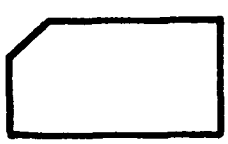
Subroutine



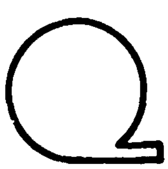
Inline Computations



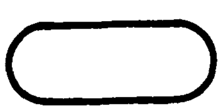
Inline Printing



Inline Card Punching or reading



Read or Write Auxiliary
Storage Device (Tape or Disk)

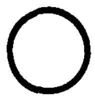


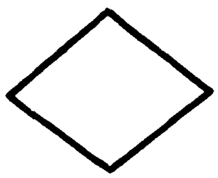
Return from Subroutine to Main Flow

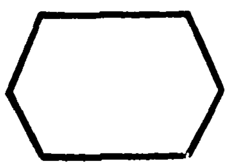
Preceding page blank

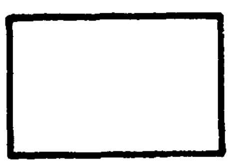
APPENDIX A

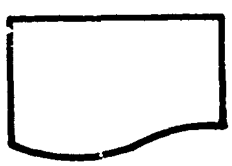
FLOW CHART NOMENCLATURE

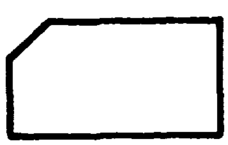

Statement Number

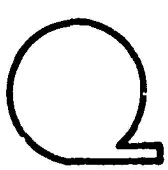

Decision

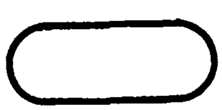

Subroutine


Inline Computations


Inline Printing

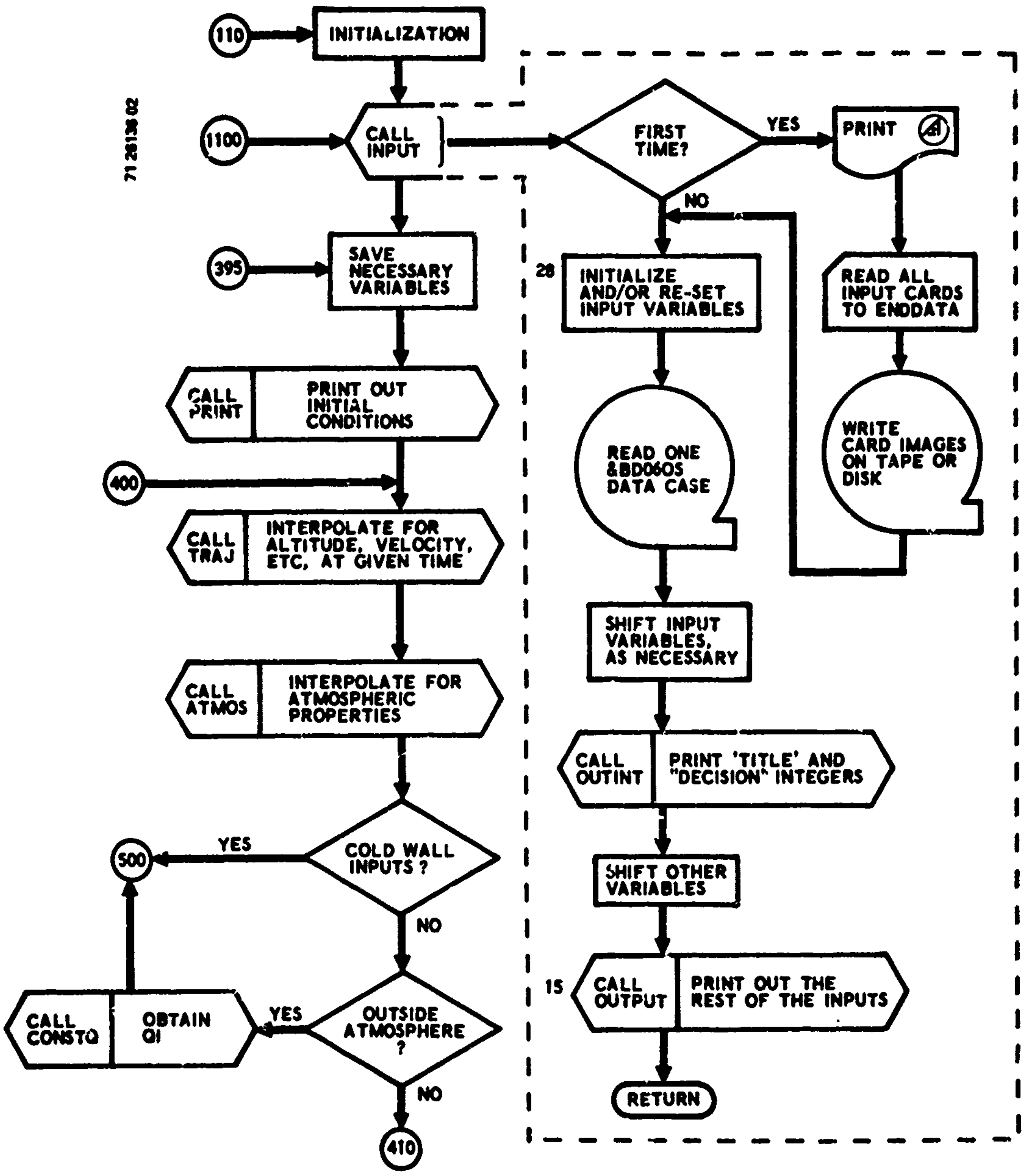

Inline Card Punching or reading


Read or Write Auxiliary
Storage Device (Tape or Disk)

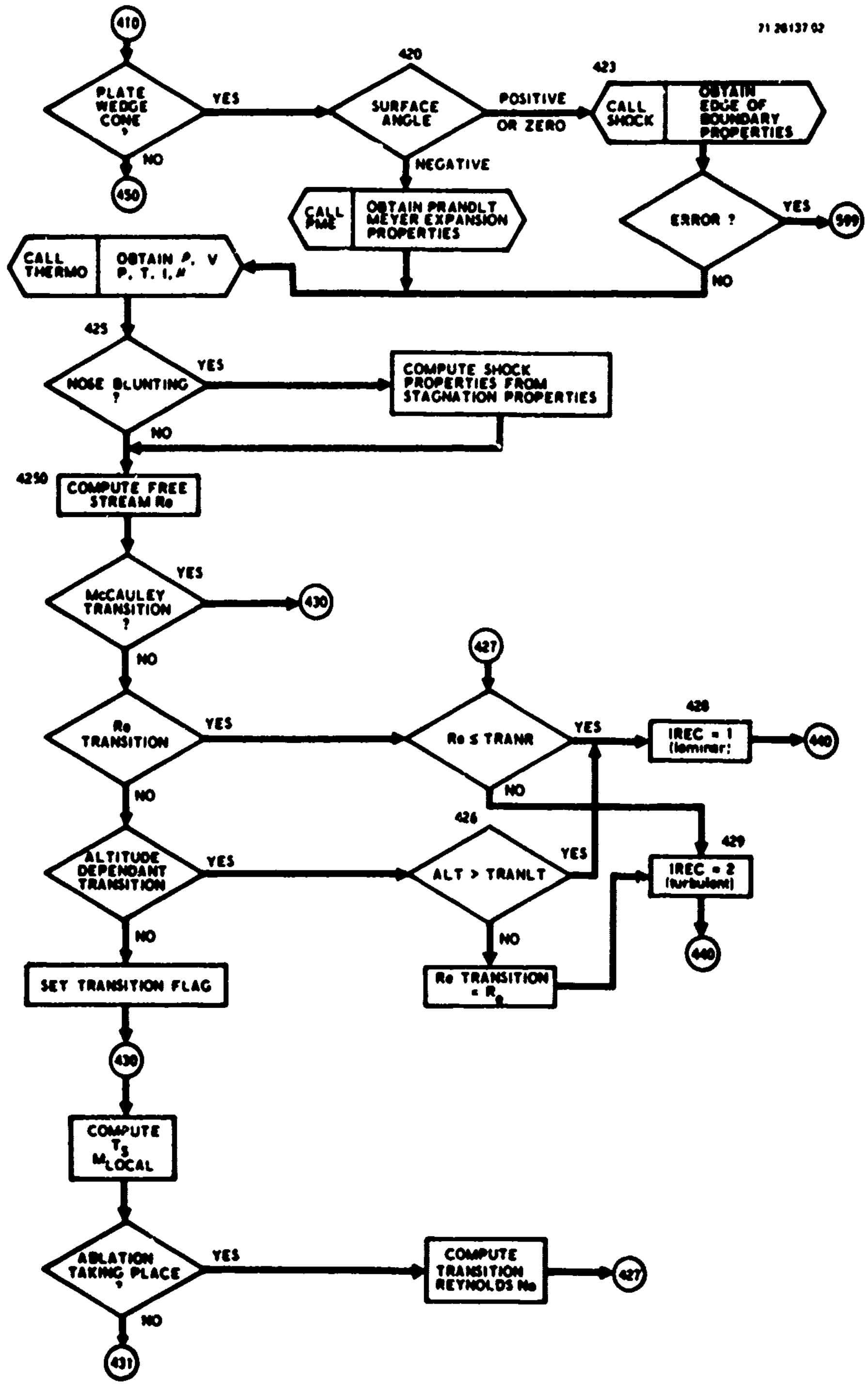

Return from Subroutine to Main Flow

Preceding page blank

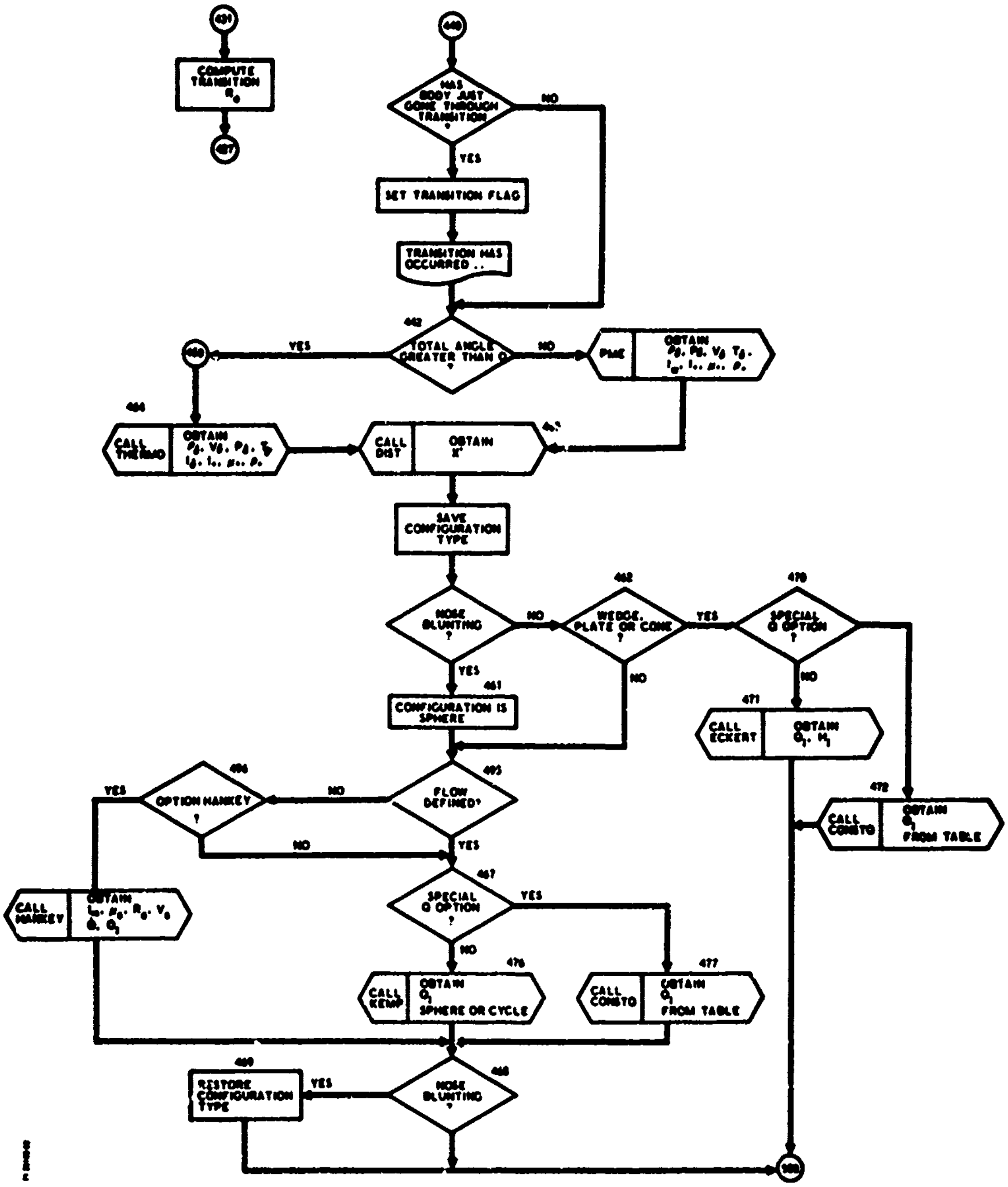
MAIN



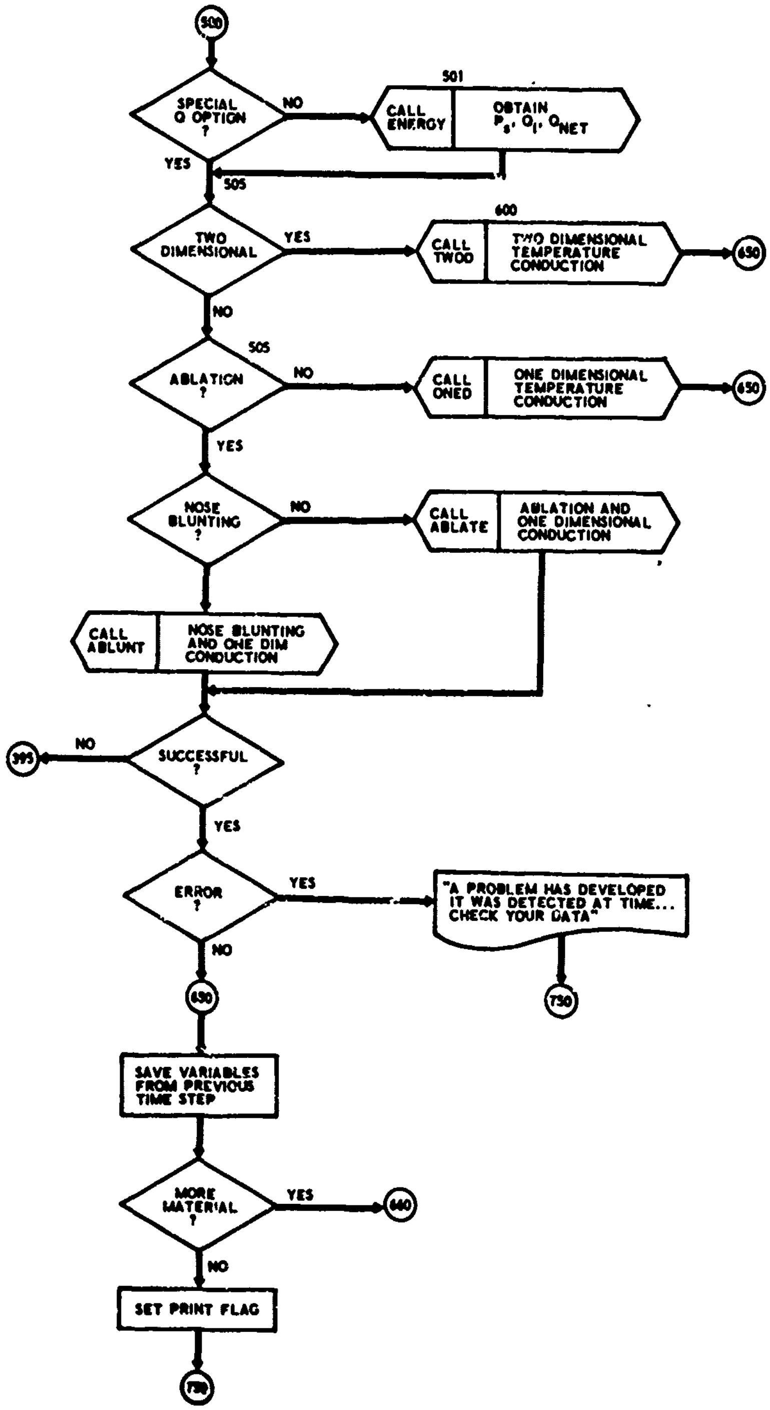
110
INITIALIZATION
1100
CALL INPUT
FIRST TIME?
YES
PRINT
NO
395
SAVE NECESSARY VARIABLES
28
INITIALIZE AND/OR RE-SET INPUT VARIABLES
READ ALL INPUT CARDS TO ENDDATA
CALL PRINT
PRINT OUT INITIAL CONDITIONS
READ ONE &BD060S DATA CASE
WRITE CARD IMAGES ON TAPE OR DISK
400
CALL TRAJ
INTERPOLATE FOR ALTITUDE, VELOCITY, ETC, AT GIVEN TIME
SHIFT INPUT VARIABLES, AS NECESSARY
CALL ATMOS
INTERPOLATE FOR ATMOSPHERIC PROPERTIES
CALL OUTINT
PRINT 'TITLE' AND "DECISION" INTEGERS
SHIFT OTHER VARIABLES
500
YES
COLD WALL INPUTS ?
NO
15
CALL OUTPUT
PRINT OUT THE REST OF THE INPUTS
CALL CONSTQ
OBTAIN QI
YES
OUTSIDE ATMOSPHERE ?
NO
RETURN
410
71 28138 02

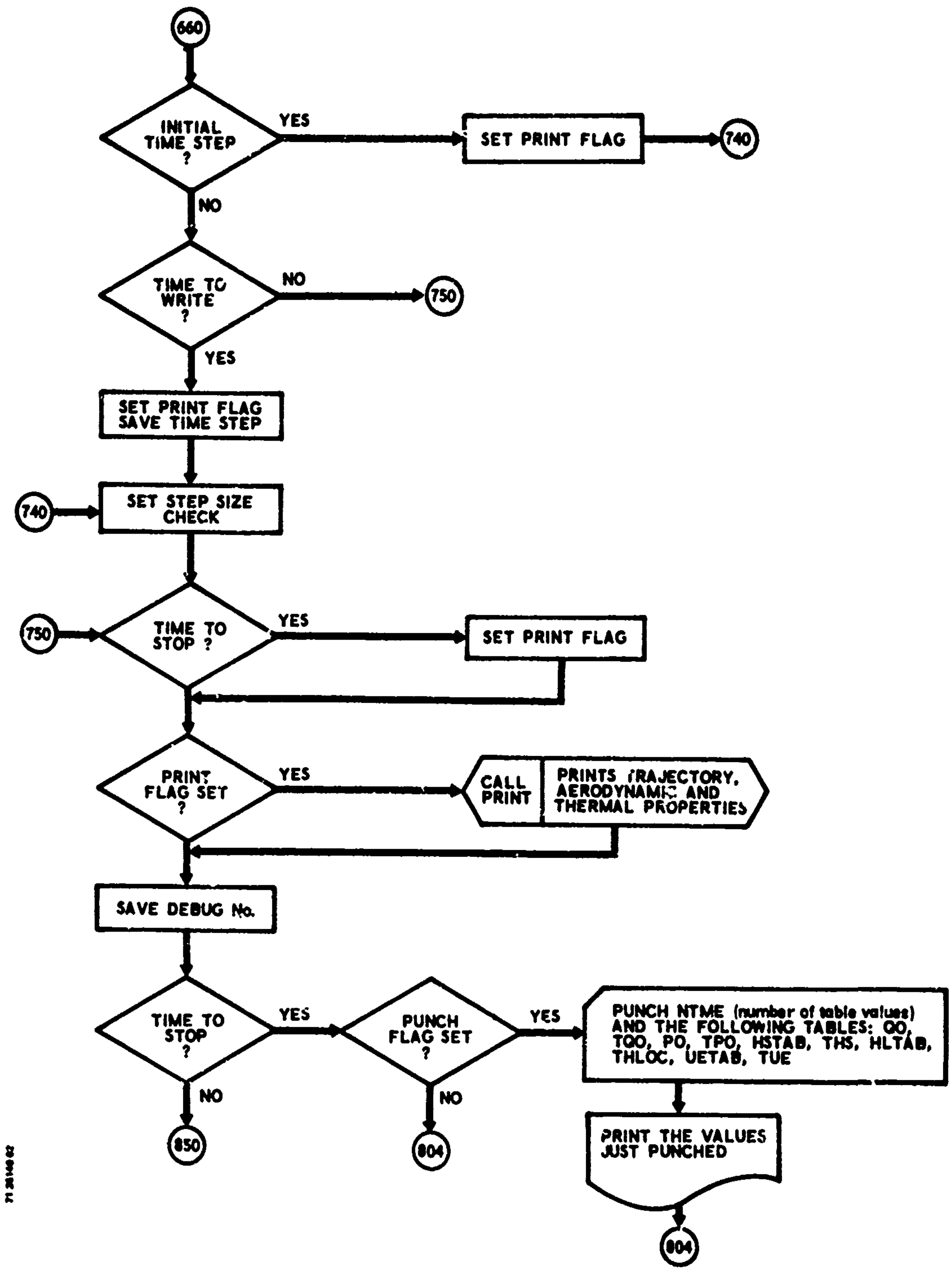


71 2613702
410
PLATE WEDGE CONE ?
YES
NO
450
420
SURFACE ANGLE
POSITIVE OR ZERO
NEGATIVE
423
CALL SHOCK
OBTAIN EDGE OF BOUNDARY PROPERTIES
ERROR ?
YES
599
NO
CALL PME
OBTAIN PRANDLT MEYER EXPANSION PROPERTIES
424
CALL THERMO
OBTAIN ρ, V P, T, I, μ
425
NOSE BLUNTING ?
YES
COMPUTE SHOCK PROPERTIES FROM STAGNATION PROPERTIES
NO
4250
COMPUTE FREE STREAM Re
McCAULEY TRANSITION ?
YES
430
NO
427
Re TRANSITION
YES
Re ≤ TRANR
YES
428
IREC = 1 (laminar)
440
NO
426
429
ALTITUDE DEPENDANT TRANSITION
YES
ALT > TRANLT
YES
IREC = 2 (turbulent)
440
NO
NO
SET TRANSITION FLAG
Re TRANSITION = R
430
COMPUTE T
M LOCAL
ABLATION TAKING PLACE ?
YES
COMPUTE TRANSITION REYNOLDS No
427
NO
431

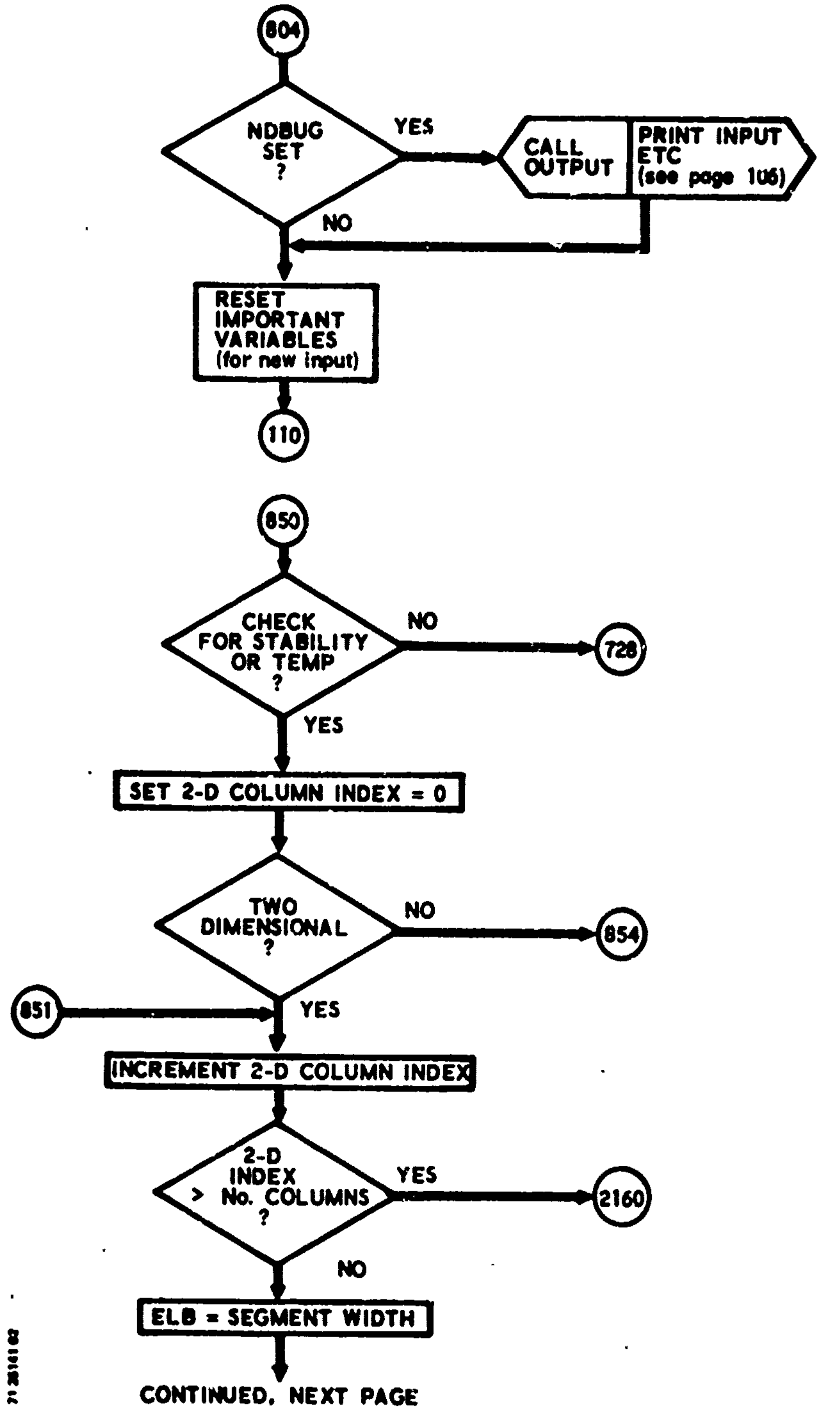


COMPUTE TRANSITION
HAS BODY JUST GONE THROUGH TRANSITION ?
NO
YES
SET TRANSITION FLAG
TRANSITION HAS OCCURRED ..
TOTAL ANGLE GREATER THAN 0 ?
YES
NO
OBTAIN
CALL THERMO
OBTAIN
CALL DIST
OBTAIN
SAVE CONFIGURATION TYPE
NOSE BLUNTING ?
NO
WEDGE, PLATE OR CONE ?
YES
SPECIAL Q OPTION ?
YES
CONFIGURATION IS SPHERE
NO
NO
CALL ECKERT
OBTAIN
OPTION HANKEY ?
NO
FLOW DEFINED?
YES
NO
YES
CALL CONSTQ
OBTAIN
FROM TABLE
CALL HANKEY
OBTAIN
SPECIAL Q OPTION ?
YES
NO
CALL KEMP
OBTAIN
SPHERE OR CYCLE
CALL CONSTQ
OBTAIN
FROM TABLE
RESTORE CONFIGURATION TYPE
YES
NOSE BLUNTING ?

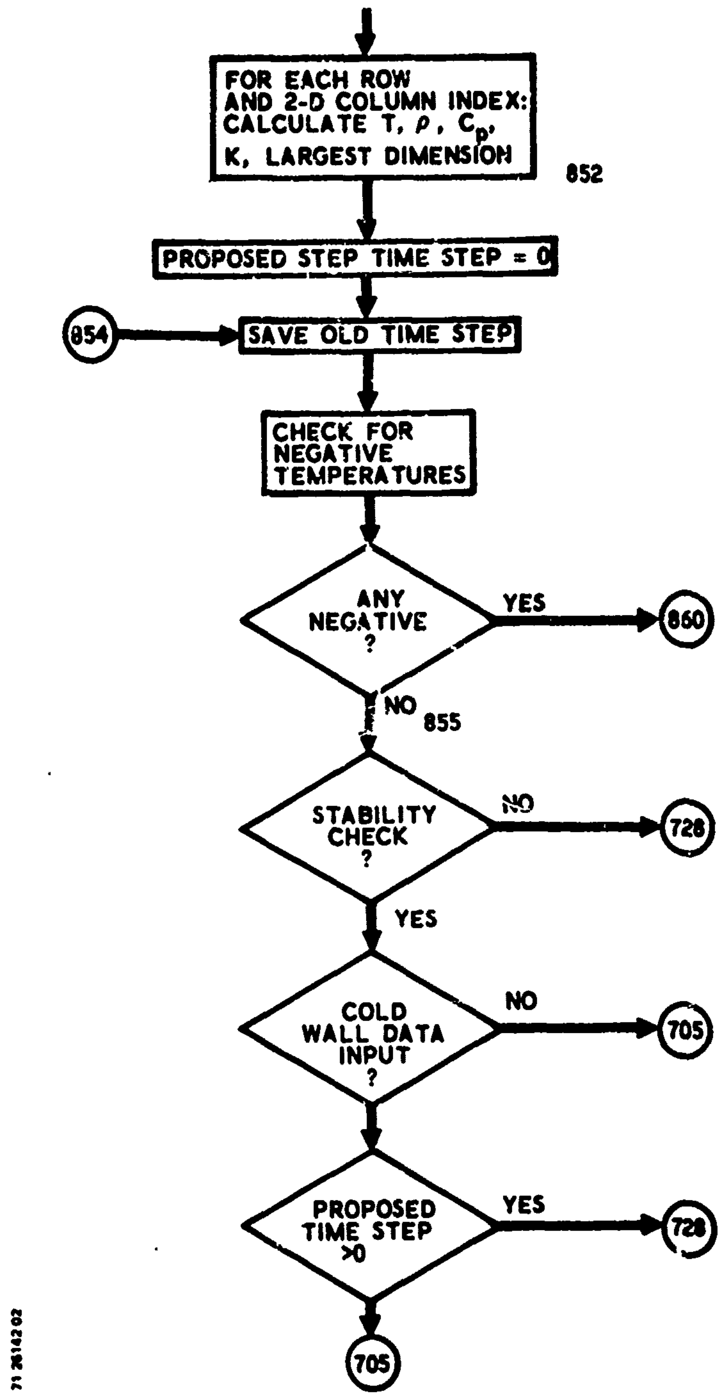

500
SPECIAL Q OPTION ?
NO
YES
501
CALL ENERGY
OBTAIN P_s, Q_I, Q_{NET}
505
TWO DIMENSIONAL
YES
600
CALL TWOD
TWO DIMENSIONAL TEMPERATURE CONDUCTION
650
NO
505
ABLATION ?
NO
CALL ONED
ONE DIMENSIONAL TEMPERATURE CONDUCTION
650
YES
NOSE BLUNTING ?
NO
CALL ABLATE
ABLATION AND ONE DIMENSIONAL CONDUCTION
CALL ABLUNT
NOSE BLUNTING AND ONE DIM CONDUCTION
395
NO
SUCCESSFUL ?
YES
ERROR ?
YES
"A PROBLEM HAS DEVELOPED IT WAS DETECTED AT TIME... CHECK YOUR DATA"
730
NO
630
SAVE VARIABLES FROM PREVIOUS TIME STEP
MORE MATERIAL ?
YES
640
NO
SET PRINT FLAG
730

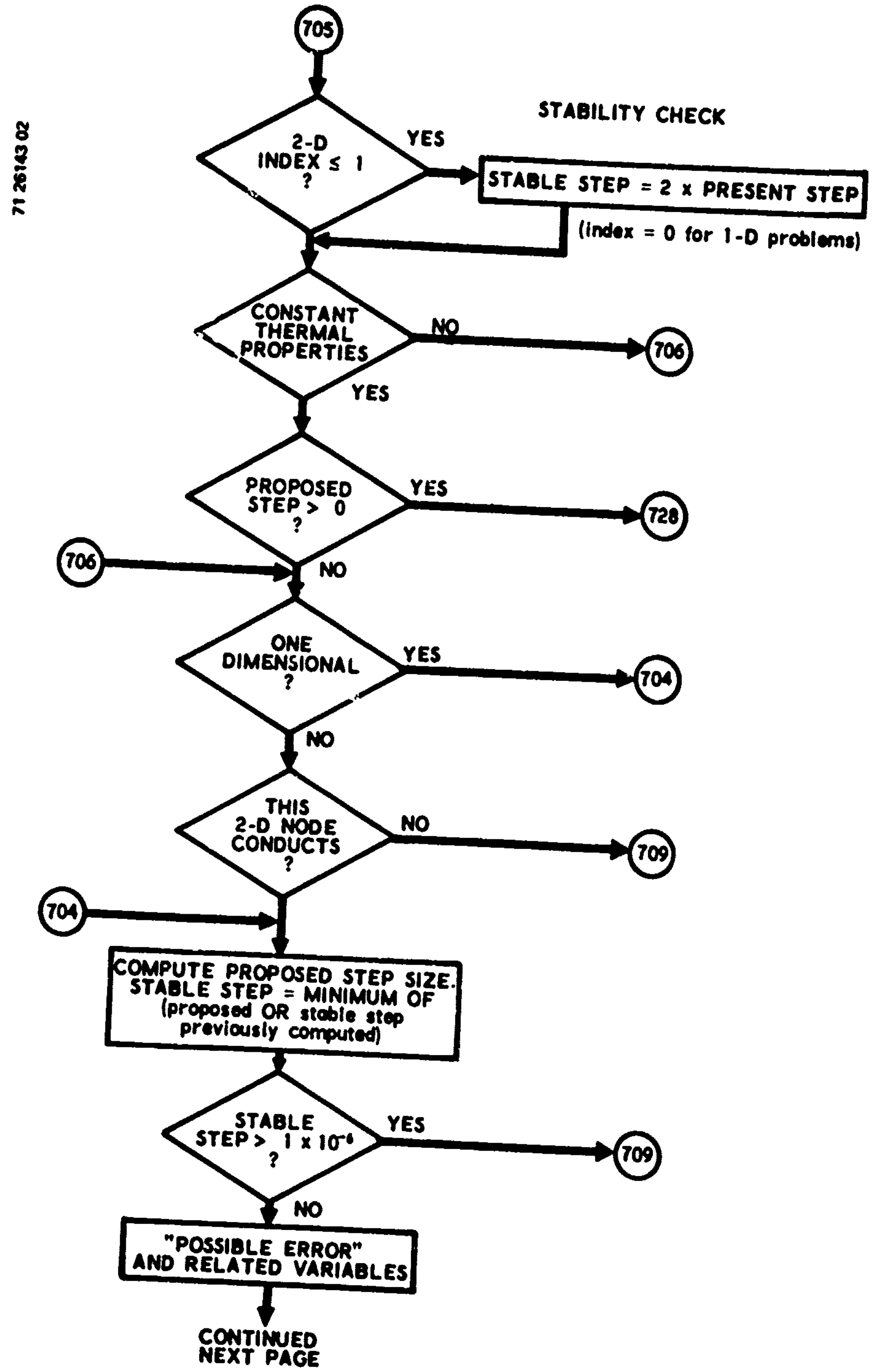


660
INITIAL TIME STEP ?
YES
SET PRINT FLAG
740
NO
TIME TO WRITE ?
NO
750
YES
SET PRINT FLAG SAVE TIME STEP
740
SET STEP SIZE CHECK
750
TIME TO STOP ?
YES
SET PRINT FLAG
PRINT FLAG SET ?
YES
CALL PRINT
PRINTS TRAJECTORY, AERODYNAMIC AND THERMAL PROPERTIES
SAVE DEBUG No.
TIME TO STOP ?
YES
PUNCH FLAG SET ?
YES
PUNCH NTME (number of table values) AND THE FOLLOWING TABLES: QO, TQO, PO, TPO, HSTAB, THS, HLTAB, THLOC, UETAB, TUE
NO
850
NO
804
PRINT THE VALUES JUST PUNCHED
804

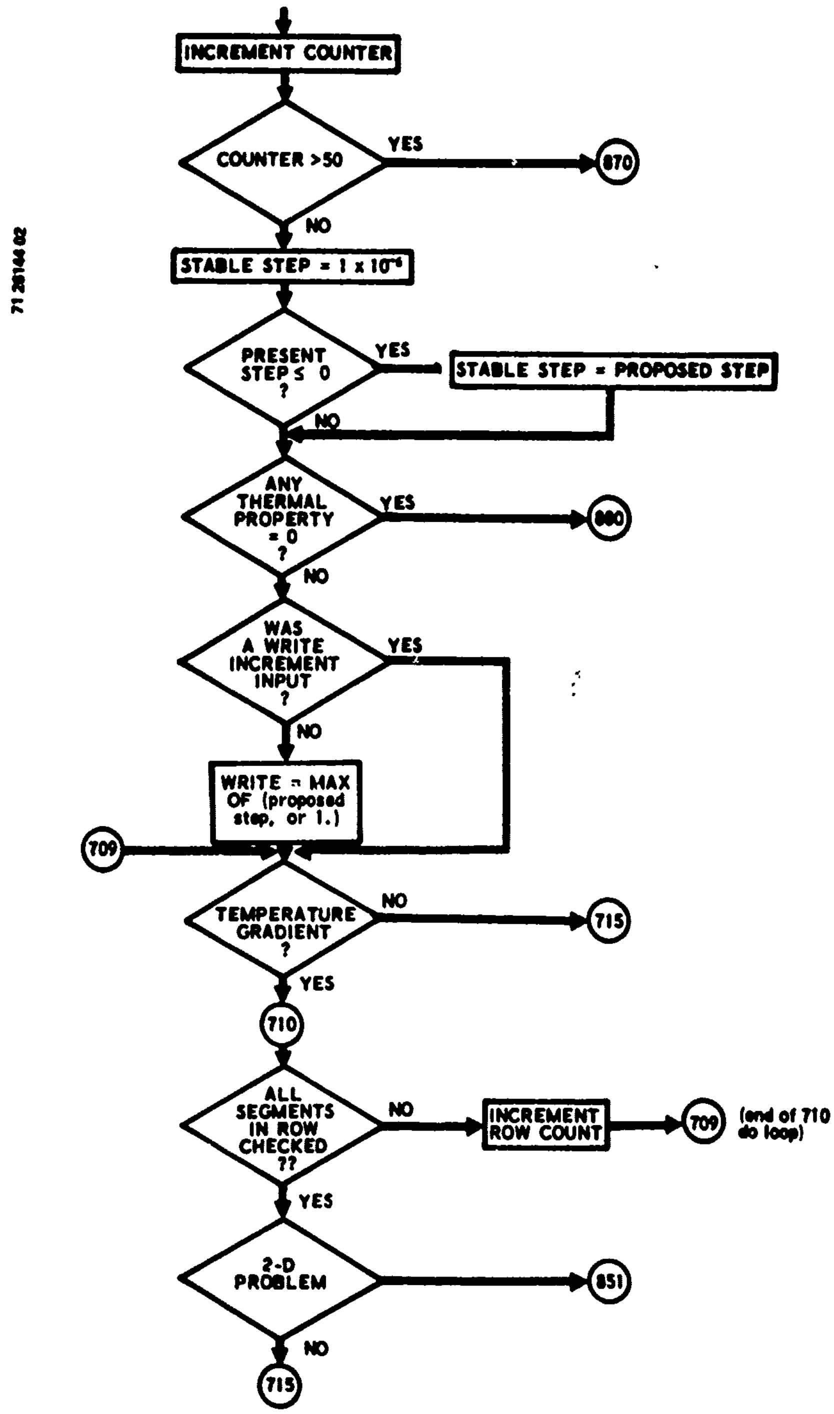

804
NDBUG SET ?
YES
CALL OUTPUT
PRINT INPUT ETC (see page 106)
NO
RESET IMPORTANT VARIABLES (for new input)
110
850
CHECK FOR STABILITY OR TEMP ?
NO
728
YES
SET 2-D COLUMN INDEX = 0
TWO DIMENSIONAL ?
NO
854
851
YES
INCREMENT 2-D COLUMN INDEX
2-D INDEX > No. COLUMNS ?
YES
2160
NO
ELB = SEGMENT WIDTH
CONTINUED, NEXT PAGE

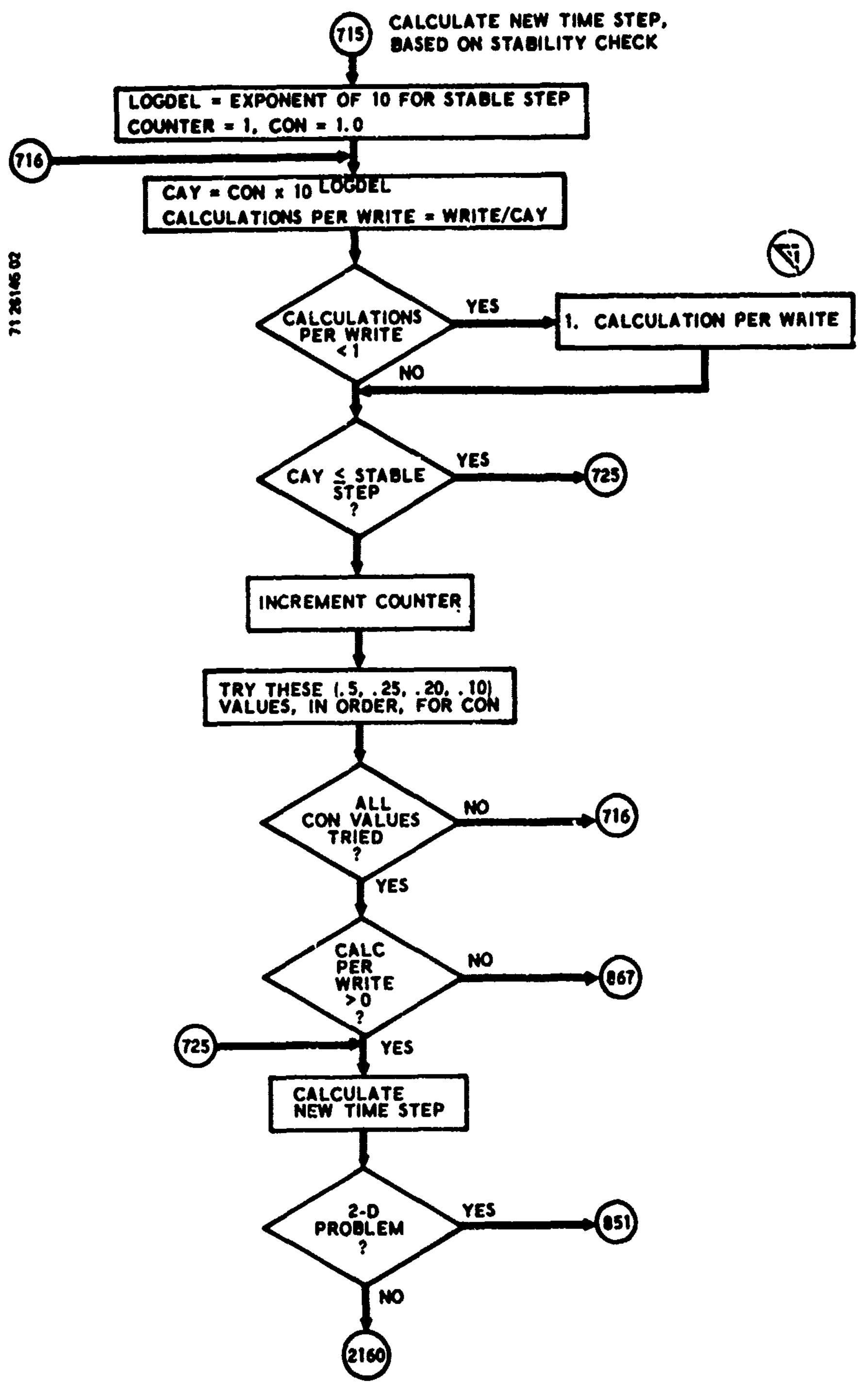
Continued from Page A-7

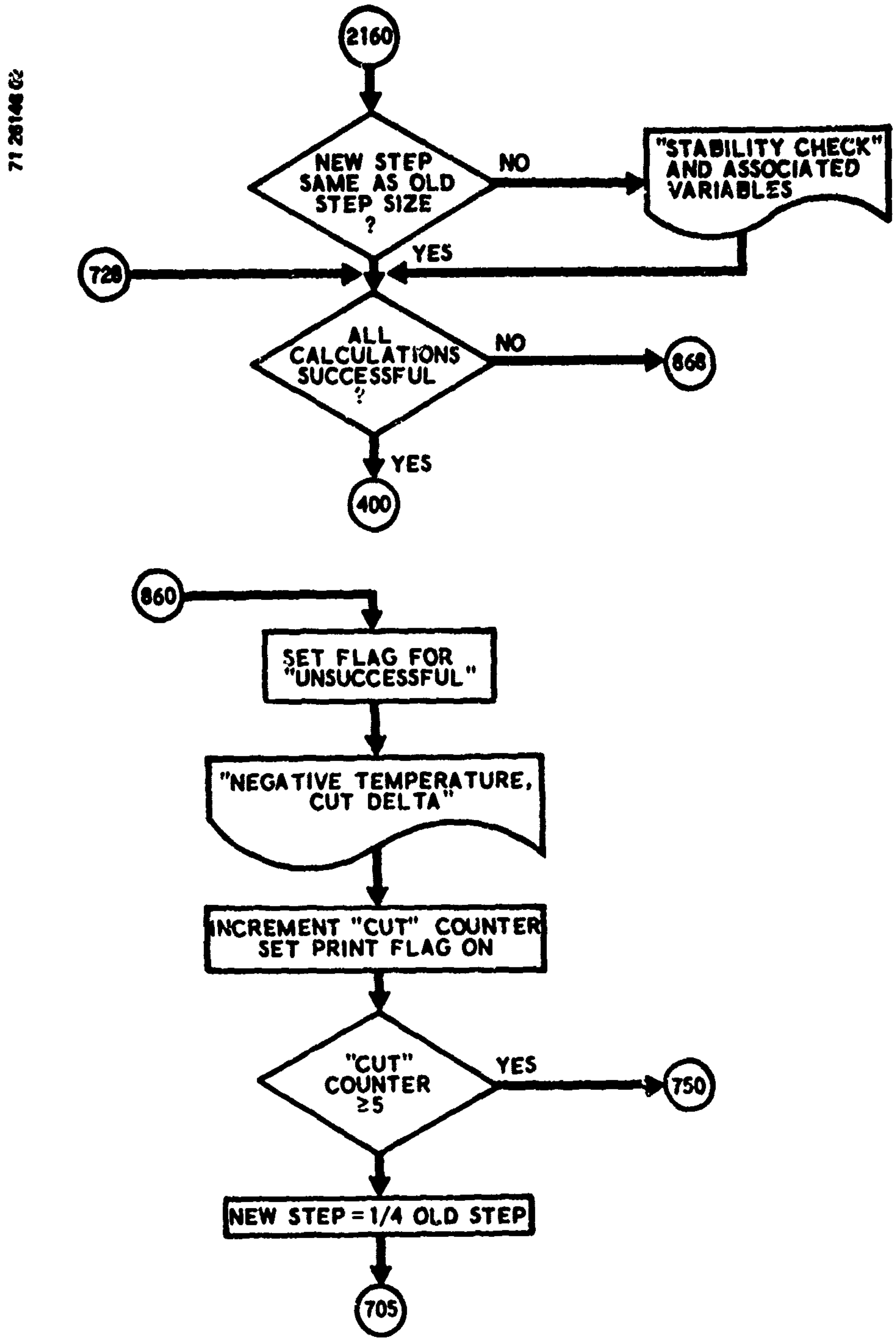


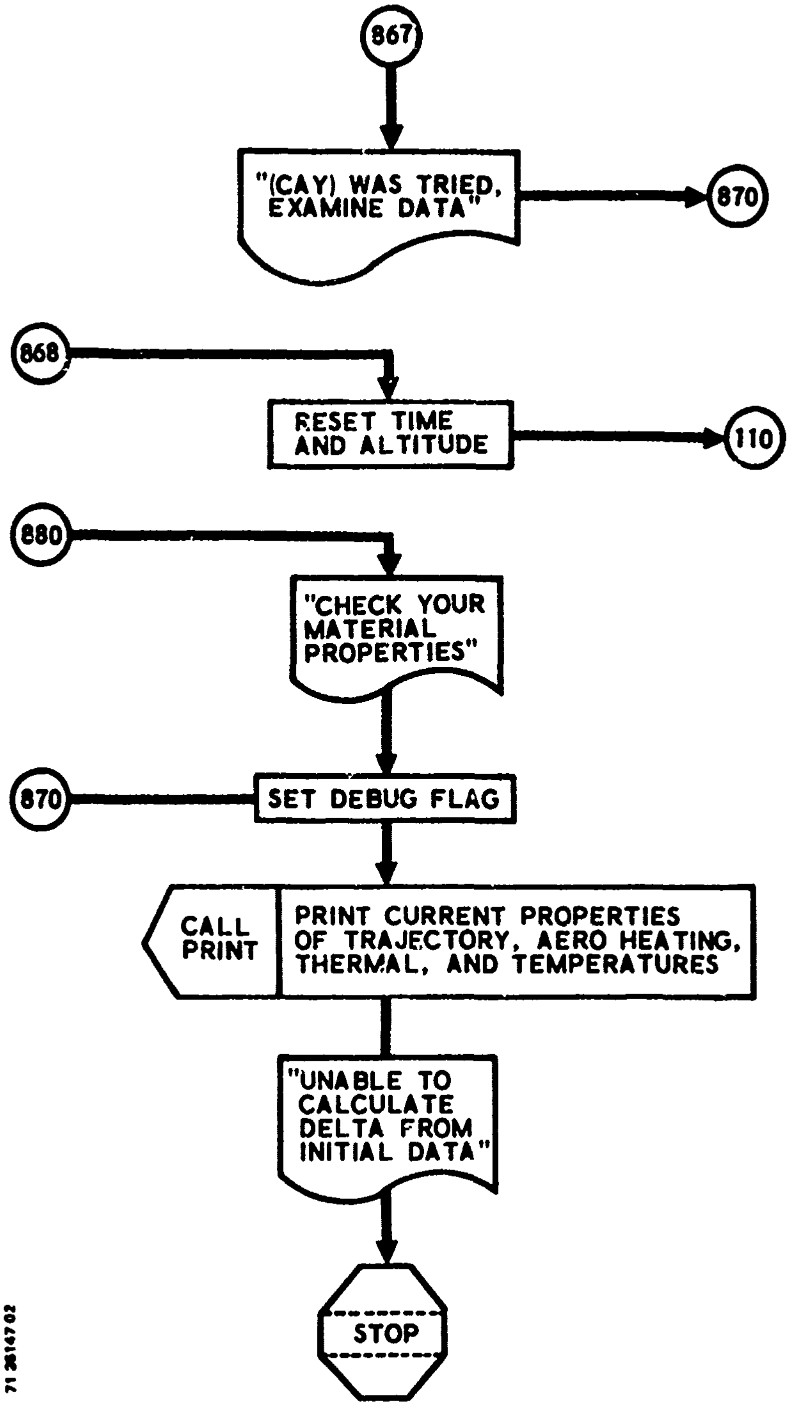

705
STABILITY CHECK
2-D INDEX ≤ 1 ?
YES
STABLE STEP = 2 x PRESENT STEP
(index = 0 for 1-D problems)
CONSTANT THERMAL PROPERTIES
NO
706
YES
PROPOSED STEP > 0 ?
YES
728
706
NO
ONE DIMENSIONAL ?
YES
704
NO
THIS 2-D NODE CONDUCTS ?
NO
709
704
COMPUTE PROPOSED STEP SIZE.
STABLE STEP = MINIMUM OF
(proposed OR stable step previously computed)
STABLE STEP > 1 x 10⁻⁶ ?
YES
709
NO
"POSSIBLE ERROR" AND RELATED VARIABLES
CONTINUED NEXT PAGE
71 2614302

Continued from Page A-9




715
CALCULATE NEW TIME STEP, BASED ON STABILITY CHECK
LOGDEL = EXPONENT OF 10 FOR STABLE STEP
COUNTER = 1, CON = 1.0
716
CAY = CON x 10 LOGDEL
CALCULATIONS PER WRITE = WRITE/CAY
71 26146 02
CALCULATIONS PER WRITE < 1
YES
NO
1. CALCULATION PER WRITE
CAY ≤ STABLE STEP ?
YES
725
INCREMENT COUNTER
TRY THESE (.5, .25, .20, .10) VALUES, IN ORDER, FOR CON
ALL CON VALUES TRIED ?
NO
716
YES
CALC PER WRITE > 0 ?
NO
867
725
YES
CALCULATE NEW TIME STEP
2-D PROBLEM ?
YES
851
NO
2160


2160
NEW STEP SAME AS OLD STEP SIZE ?
NO
"STABILITY CHECK" AND ASSOCIATED VARIABLES
YES
728
ALL CALCULATIONS SUCCESSFUL ?
NO
868
YES
400
860
SET FLAG FOR "UNSUCCESSFUL"
"NEGATIVE TEMPERATURE, CUT DELTA"
INCREMENT "CUT" COUNTER SET PRINT FLAG ON
"CUT" COUNTER ≥5
YES
750
NEW STEP = 1/4 OLD STEP
705
71 28148 G


867
"(CAY) WAS TRIED, EXAMINE DATA"
870
868
RESET TIME AND ALTITUDE
110
880
"CHECK YOUR MATERIAL PROPERTIES"
870
SET DEBUG FLAG
CALL PRINT
PRINT CURRENT PROPERTIES OF TRAJECTORY, AERO HEATING, THERMAL, AND TEMPERATURES
"UNABLE TO CALCULATE DELTA FROM INITIAL DATA"
STOP

71 2614702

APPENDIX B

Program Listing

Preceding page blank

```
C       OSBD06  DECK             MAIN PROG. FOR AEROSPACE AERO-HEATING      BD06    0
COMPILATIONS HAVE BEEN MADE UNDER FORTRAN (G.AND.H*2)
CARD DECK FOR THIS PROGRAM RESIDES ON AEROSPACE TAPE = 06932
C - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
CAUTION - ALTHOUGH ALL DESCRIPTIONS AND CODING IS BELIEVED TO BE CORRECT --4/15/71
C         AT THIS TIME, COMPLETE ACCURACY CANNOT BE GUARANTEED.  ALSO,
C         THIS PROGRAM IS SUBJECT TO CHANGE AT ANY TIME, (AFTER 4/15/71)
C         AS TO SPECIFICATIONS, OPTIONS, ECT., WITHOUT NOTICE
C         OR OBLIGATION. ( IT IS VERY HARD TO CHECK EACH, AND EVERY,
C         OPTION)  THIS HAS JUST BEEN A WORD OF CAUTION, SINCE IT WORKS.
C - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
COMMENT - LATEST -OPUS- LOAD MODULE (4/15/71)    A422.ASAH           --4/15/71
C          A  REFERENCE ENTHALPY/ SPACE PLANE      METHOD    ,  AUG 1967   BD06    1
C         AA  SEE- G.D.A. REPORT  AE-62-0497 BY L.FISHER  (5-24-62)        BD06    2
C        AAA  PROGRAMMED BY J.KOHLENBERGER  -MATH AND COMPUTATIONS CENTER BD06    3
C       A  A  RE-ENGINEERED BY M.GYFTVAY    -FLUID MECHANICS DEPT.         BD06    4
C     A A AEROSPACE CORPORATION - SAN BERNARDINO OPPERATIONS -  CALIF.     BD06    5
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)BD06E   5
     1   ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)     BD06E   6
     2   ,   QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)    BD06E   7
     3   ,   A(4,40),B(4,40)                                             BD06E   8
     4   ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)      BD06E   9
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI BD06E  10
     1   ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                  BD06E  11
     2   ,   AA, BB, CC,CP2, DK, T2                                      BD06E  12
      COMMON       ALT,AMINF,AOFA,ANGLE, CCNPR, DELTA,DQSAV              BD06E  13
     1   ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                        BD06E  14
     2   ,ICHK,ICK,ICONF,IGO,INB4,IN342,IREC,          KASE,KFUNT,    NPQBD06E  15
     3   ,   NBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 BD06E  16
     4   ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   BD06E  17
     5   ,   RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLBD06E  18
     6   ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR    BD06E  19
     7   ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE        BD06E  20
     8   ,   XX,XDIST                                                    BD06E  21
     9      ,ETATP,ETALP,HVPSTM,PSTMX,NT
      EQUIVALENCE (NSEG,NAROW),(QCALT,TRANLT)
      COMMON /BLOK1/ TITLE(20)                                           BD06E  23
      COMMON /CHARAB/ NTME,TSTOR(50),QO(50),PO(50),HSTAB(50),HLTAB(50),  BD06G  24
     1 UETAB(50)                                                         BD06E  25
      COMMON / SWALOW/  QINT(50),HEDGE(50),PEDGE(50),UEDGE(50),HRECOV(50BD06E  26
     1)  ,  INPCW ,ISET                                                  BD06E  27
C          FOR ARCHIE OSSIN - COLD WALL INPUTS FROM ENTROPY SWALLOWING   BD06E  28
      COMMON / MIKE / HV2,ETAL2,ETAT2,XI, FV1P(10),FV1S(10),FV2P(10),    BD06E  29
                                                                         BD06E  30
```

```
     *       FV2S(10),STGTBL(10),SHRTBL(10),NOPSTG,NOPSHR                    BD06E 31
     2   ,   QSONIC,SHEAR,HOABLM,HF, ISTETS,  NOCHEK,                        INPT  34
     3       RHOTBL(10,40),CPTBL(10,40),EKTBL(10,40),TEMTBL(10,40)           BD06E 33
      COMMON /WERNER/   TRVRSL,TFB(100),RHOFB(100),PFB(100),ALTCM(100)
      DOUBLE PRECISION   FLOW(3)
      DATA  FLOW/8HLAMINAR ,8HTURBULNT,8HUNKNOWN /
      DATA ALTOLD,RADOLD,TIMOLD,VINOLD,MTRANS/3*0.,.1E-8,0/
      NAMELIST /TRAJLT/ NTRAJ,TIME,ALT,VINF,AOFA, DELTA,WRITE,STOP,
     1 ICK,TINF,PINF,RHOINF, ICONF,ANGLE
      NAMELIST /TROUBL/ A,ALT1,AOFA1,VAPRES ,ALTIO
      NAMELIST/RNFIND/  QNET,HOABL,QSUBI,DELTA ,WRITE,STOP,HSUBI,ICONF,I
     1GO,RADOLD,RADIUS,HF,XX,XDIST,WAB, XI,NDBUG,ISFT,ISTETS,TIME,ALT
      T(SR) = 17.899 + 4.0675*SR -.33884E-4*(SR*SR) -.3495E-6*SR*(SR*SR)BD06E 34
CALL TO ABCLOS SETS UP ERROR TRAPS FOR ADDRESSING TYPE ERRORS, ECT.
      CALL ABCLOS
C            TO INITIALIZE BLANK COMMON
      CALL ZERBLK
C            TO INITIALIZE BLANK COMMON
      TRVRSL=0.0
      IGO = -1
  110 VSHOLD = 0.
      RHSOLD = 0.
      SMUOLD = 0.
      SHIOLD = 0.
      QVIOLD = 0.
      PSHOLD = 0.
      RCIOLD =0.
      QCCOLD =QCONS
      TEGOLD = TWALL
C     XDIOLD = XDIST
      XDIOLD = XX
C                                           INPUT
 1100 CAL   INPUT
C                                           INPUT
      DELATH=.1E-8
      IF ((NSFG.EQ.1).OR.(NDIM.EQ.2))  DELATH=.00001
      DELBPT = -2.
      HOABLM = 0.                                                         BD06E 71
      HSUBI=0.                                                            BD06E 62
      IDELFR = 0
      TRCOLD = 3
      IREC = 3                                                                  66
      ISET = 0
      MTRANS = 0                                                          BD06E 66
      NOBUG2 = NOBUG                                                      BD06E 68
      QCCOLD = QCONS
```

```
      QNET =0.                                                      BD06E 61
      QSONIC = 0.                                                   BD06E 69
      QSUBI = 0.0
      RADOLD = RADIUS
      RECI =0.                                                      BD06E 63
      REFI = 0.
      REYN = 0.                                                     BD06E 60
      SCON = NDIM
      SCON = SCON*12.5
      SHEAR = 0.                                                    BD06E 70
      SHKI=0.                                                       BD06E 65
      VSHK=0.                                                       BD06E 64
      IF (ALT) 111,112,111                                          BD06E 72
  111 TIME = ALT                                                    BD06E 73
      GO TO 113                                                     BD06E 74
  112 TIME = TIME1(1)                                               BD06E 75
  113 NDIM = NDIM                                                   BD06F 76
C     INITIALIZE PARAMETERS                                         BD06E 77
      INB4 = 0                                                      BD06E 78
  120 QSUM = 0.0                                                    BD06E 79
      QINSD = 0.0                                                   BD06E 80
  125 QSAV = 0.0                                                    BD06E 81
      IGO  = 0                                                      BD06E 82
      WAB  = 0.0                                                    BD06E 83
      WABDOT = 0.0                                                  BD06E 84
      NSEG1 = 1                                                     BD06E 85
  130 NSEG2 = 1                                                     BD06E 86
      GO TO (140, 180 ), NDIM                                       BD06E 87
  140 TWALL = TEMP(1)                                               BD06E 88
      GO TO 300                                                     BD06E 89
  180 TWALL = T2(1,1)                                               BD06E 90
  300 DELSAV = DELTA                                                BD06E 91
      IF (NOSB.LE.0)   GO TO 396
C     IF (NDBUG) 396,400,396                                        BD06E 92
  395 ALT = ALTOLD                                                  BD06E 93
      WRITE (6,RNFIND)
      PSHK   = PSHOLD                                               BD06E102
      QCONS  = QCCOLD                                               BD06E104
      QSUBI  = QVIOLD                                               BD06E101
      RADIUS = RADOLD                                               BD06E106
      RECI   = RCIOLD                                               BD06E103
      RHOSHK = RHSOLD                                               BD06E 97
      SHKI   = SHIOLD                                               BD06E100
      SHKMU  = SMUOLD                                               BD06E 99
      IF (ISET.GT.0) TIME = TIMOLD                                  BD06E 95
      TWALL  = TEGOLD                                               BD06E105
```

```
              VINF = VINOLD                                                    BD06E 94
              VSHK    = VSHOLD                                                 BD06E 96
      C       XDIST   = XDIOLD                                                 BD06E 98
              XX      = XDIOLD
          396 IF (NDBUG.GT.1)
             1WRITE(6,TROUBL)
              IF (NDIM.EQ.1)    CALL GETCPK
              IF (NDIM.EQ.2)    CALL GECPK2
              WRITE (6,399)
          399 FORMAT (' INITIAL CONDITIONS')
              CALL PRINT                                                       BD06E107
              WRITE (6,399)
          400 CALL TRAJ                                                        BD06E108
              CALL ATMOS                                                       BD06E109
              TRANSF = 0.
      C             ICK = 1      FROM -ATMOS-                                  BD06E111
      C     OR     ICK = 0 , IF OUTSIDE ATMOSPHERE (.GT. 297000.-FT.)
              IF (INPCW.GE.1) GO TO 500                                        BD06E110
              IF (NDBUG.GE.1) WRITE (6,TRAJLT)
              IF (ICK ) 410, 405, 410                                          BD06E112
          405 CALL CONSTQ                                                      BD06E113
              GO TO 500                                                        BD06E114
      C             IREC ,       1=LAMINAR,  2=TURBULENT,  3=CONDITIONS UNKNOWN  BD06E115
          410 IREC = 3                                                         BD06E116
          415 IF ( ICONF-3 ) 420, 420, 450                                     BD06E117
      C             ICONF =      FLAT PLATE,   WEDGE,   OR CONE                BD06E118
          420 IF(ADFA+ANGLE) 422,423,423                                       BD06E119
          422 CALL PME                                                         BD06E120
              GO TO 425                                                        BD06E121
          423 CALL SHOCK                                                       BD06E122
              IF (IGO)  599,424,599                                            BD06E123
          424 CALL THERMO                                                      BD06E124
          425 IF (NOSB.LL.0) GO TO 4250                                        BD06E125
      C             HYPERSONIC  FLOW  NOSE  BLUNTING  CORRECTIONS   5/13/67 MJG  BD06E126
              PSTAG = PINF*(1.2*(AMINF**2))**3.5 *(6./(7.*(AMINF**2)-1.))**2.5  BD06E127
             1   / 2116.                                                       BD06E128
              PSHK =1160. *PSTAG                                               BD06E129
              VSHK =.301 * VINF                                                BD06E130
              RHOSHK=7.24 *RHOINF                                              BD06E131
              SHKMU=1.511E-10 * VINF                                           BD06E132
              SHKI = STAGI- VSHK**2/5.01E+4                                    BD06E133
              XDIST = RADIUS                                                   BD06E134
         4250 REYN = VSHK*RHOSHK*XDIST / SHKMU                                 BD06E135
              IF (REYN.LE.0.)  WRITE(6,4252)  VSHK,RHOSHK,XDIST,SHKMU
         4252 FORMAT ('  REYN ERROR, ' 4F16.8)
      C             G.E. CRITERIA      TRANSITION                              BD06E136
```

```
      IF (MTRANS.GT.0) GO TO 430                                        BD06E137
C           REYNOLDS NO.       TRANSITION                               BD06E138
      IF (TRANR.GT.0.) GO TO 427                                        BD06E139
C           ALTITUDE DEPENDANT TRANSITION                               BD06E140
      IF (TRANLT.GT.0.) GO TO 426                                       BD06E141
      MTRANS = 1                                                        BD06E142
      FT = 1.0                                                          BD06E143
      GO TO 430                                                         BD06E144
  426 IF((ALT-TRANLT).GT.0.) GO TO 428                                  BD06E145
      TRANR = REYN                                                      BD06E146
      GO  TO 429                                                        BD06E147
  427 IF (REYN-TRANR) 428,428,429                                       BD06E148
  428 IREC = 1                                                          BD06E149
      GO TO 440                                                         BD06E150
  429 IREC = 2                                                          BD06E151
      GO TO 440                                                         BD06E152
C           430   TSHK  FIX TO EXTEND FOR HIGHER SHKI VALUES 5-4-67     BD06E153
  430 BIGH = SHKI / 33.86                                               BD06E154
      PEXP = ALOG10(PSHK/2116.)                                         BD06E155
      TEA  = 230.63/(10.-PEXP)+ .18304*(PEXP+3.00)                      BD06E156
      THETA= 2.1965 + .31961*(PEXP+4.1)**2                              BD06E157
      D    = (THETA-TEA)/(2.*TEA-THETA-79.4)                            BD06E158
      SMALB= 79.4 + (1.+D)*(TEA-79.4)                                   BD06E159
      SMALA= D*(79.4-SMALB)                                             BD06E160
      C    = -SMALA*D                                                   BD06E161
      F    = -2.1965 + 1.46434/(EXP(-2.*PEXP)+1.)                       BD06E162
      GMEN = .00065  - .012096/ (PEXP+9.6)                              BD06E163
      SMALM= PEXP*(-1.6)*(1.-.5*PEXP) - 94.2                            BD06E164
      TAU1 = (5.4913-.56743*(PEXP+1.75)**2) *EXP(BIGH/50.-10.)          BD06E165
      TAW  = F*EXP(GMEN*(H+SMALM)**2)                                   BD06E166
      TRAT = SMALA+SMALB*(BIGH/250.)+C/((BIGH/250.)+D)+TAU1+TAW         BD06E167
      TSHK = 492.* TRAT                                                 BD06E168
CHANGE -   'H' WAS FOUND TO BE UNDEFINED (4/21/70), SO THE FOLLOWING IS
CHANGE -   A TEMPORARY FIX (4/29/70)
      TSHK = 4.*SHKI
CHANGE -   DAN NOWLAN CLAIMS, 'THIS IS GOOD TO ABOUT 5000'
      NAMELIST /TSHOCK/ TSHK,TRAT,TAW,TAU1,SMALM,GMEN,F,C,SMALA,SMALB,  BD06E169
     *  D,THETA,TEA,PEXP,BIGH,PSHK,SHKI                                 BD06E170
      IF((TSHK.LE.0.).OR.(NDBUG.EQ.1)) WRITE (6,TSHOCK)                 BD06E171
      AMLOC = VSHK / (49.07*SQRT(TSHK))                                 BD06E172
      IF (WABDOT.GT.0.) GO TO 431                                       BD06E173
      TRANR = 220000.* AMLOC**1.90/FT                                   BD06E174
      GO TO 427                                                         BD06E175
  431 TRANR = 75400. * AMLOC**1.891/FT                                  BD06E176
      GO TO 427                                                         BD06E177
  440 IF (IREC .EQ. IRCOLD)GO TO 442                                    BD06E178
```

```
C     CALL PRINT                                                       BD06E179
      TRANSF = 1.
      WRITE  (6,441)   FLOW(IRCOLD),FLOW(IREC),TRANLT                  BD06E181
  441 FORMAT (54H- TRANSITION HAS OCCURREC * * * * * CONDITIONS BELOW,BD06E182
     1',  FROM ',A8,'  TO  ',A8,' TRANLT=',E13.5)                     BD06E182
      IRCOLD = IREC                                                    BD06E188
  442 IF ((ANGLE +AOFA) .GE. 0.) GO TO 450                             BD06E183
  445 CALL PMF                                                         BD06E184
      GO TO 455                                                        BD06E185
C            ICONF =     SPHERE   OR   CYLINDER                        BD06E186
  450 CALL THERMO                                                      BD06E187
  455 CALL DIST                                                        BD06E189
  460 IC = ICONF                                                       BD06E190
      IF (NOSB) 462,462,461                                            BD06E191
  462 IF (ICONF-3) 470,470,495                                         BD06E192
C             FLAT PLATE, WEDGE, CONE   USE  -ECKERT-                  BD06E193
  470 IF (NPGAS) 471,471,472                                           BD06E194
  471 CALL ECKERT                                                      BD06E195
      GO TO 500                                                        BD06E196
  472 CALL CONSTQ                                                      BD06E197
      GO TO 500                                                        BD06E198
C     ICONF IS 4  WHEN SPHERICAL (BLUNT) NOSE IS USED                  BD06E199
  461 ICONF = 4                                                        BD06E200
  495 IF (IREC-2) 467,467,496                                          BD06E201
  496 IF (ICHK)   467,467,464                                          BD06E202
C            SPHERE , CYLINDER      - USE  HANKEY -                    BD06E203
  464 CALL HANKEY                                                      BD06E204
      GO TO 468                                                        BD06E205
C            SPHERE , CYLINDER      - USE  KEMP-RIDDLE -               BD06E206
  467 IF (NPGAS)  476,476,477                                          BD06E207
  476 CALL KEMP                                                        BD06E208
      GO TO 468                                                        BD06E209
  477 CALL CONSTQ                                                      BD06E210
  468 IF(NOSB) 500,500,469                                             BD06E211
  469 ICONF = IC                                                       BD06E212
  500 IF (NPGAS) 501,501,505                                           BD06E213
C           IF NPGAS + , QI FOUND IN CONSTQ AND SKIP ENERGY            BD06E214
  501 CALL ENERGY                                                      BD06E215
  505 GO TO (550,600), NDIM                                            BD06E216
  550 IF ( TABL ) 560, 560, 570                                        BD06E217
  560 CALL PNED                                                        BD06E218
      GO TO 650                                                        BD06E219
C           ABLATION  -  STANDARD OR NOSE BLUNTING                     BD06E220
  570 IF (NOSB.LE.0)    GO TO 573                                      BD06E221
  571 CALL ABLUNT                                                      BD06E222
      IF (NDBUG.GT.0)  WRITE (6,RNFIND)
```

```
      IF (ISET .GE. 1) GO TO 395                                        BD06E223
      GO TO 575                                                         BD06E224
  573 CALL ABLATE                                                       BD06E225
      IF (NSEG2.GT.NSEG) IGO=1
  575 IF (IGO)  599,650,599                                             BD06E226
  599 NDBUG = 1                                                         BD06E227
      WRITE (6,598) TIME
  598 FORMAT ('- A PROBLEM HAS DEVELOPED'/' IT WAS DETECTED AT TIME='
     1 E16.8/'-   CHECK YOUR DATA')
      TIME = STOP + DELTA
      GO TO 750
C***********************************************************************BD06E229
  600 CALL TWOD                                                         BD06E230
C***********************************************************************BD06E231
  650 DELSAV = DELSAV + DELTA                                           BD06E232
      ALTOLD = ALT                                                      BD06E233
      TIMOLD = TIME                                                     BD06E234
      VINOLD = VINF                                                     BD06E235
      VSHOLD = VSHK                                                     BD06E236
      RHSOLD = RHOSHK                                                   BD06E237
C     XDIOLD = XDIST                                                    BD06E238
      XDIOLD = XX                                                       BD06E238
      SMUOLD = SHKMU                                                    BD06E239
      SHIOLD = SHKI                                                     BD06E240
      QVIOLD = QSUBI                                                    BD06E241
      PSHOLD = PSHK                                                     BD06E242
      RCIOLD = RFCI                                                     BD06E243
      QCCOLD = QCONS                                                    BD06E244
      TFGOLD = TWALL                                                    BD06E245
      RADOLD = RADIUS                                                   BD06E246
      IF (NSEG2.LT.NSEG) GO TO 660                                      BD06E247
      IF (NSEG.EQ.1) GO TO 660
C     CALL PRINT                                                        BD06E248
      TRANSF = 1.
      GO TO 750
C     CHECK TIME TO PRINT                                               BD06E250
  660 IF ( TIME - TIME1(1) - DELTA .GT. 0.)   GO TO 680
C 670 CALL PRINT                                                        BD06E252
      TRANSF = 1.
      GO TO 740                                                              253
  680 IF ( DELSAV.GE.WRITE)   GO TO 700
      GO TO 750
C 700 CALL PRINT                                                        BD06E255
  700 TRANSF = 1.
      DELSAV = DELTA                                                    BD06E256
  740 DELBRT = 0.                                                            257
```

```
C       CHECK STOP                                                      BD06E258
    750 IF (TIME.GE.STOP)    TRANSF = 1.
        IF (TRANSF.GT.0.)    CALL PRINT
        NDBUG = NDBUG2                                                  BD06E261
        IF (TIME.LT. STOP) GO TO 950
        IF (NPU .LE. 0) GO TO 804                                       BD06E262
        NTME = NTME - 1                                                 BD06E263
        PUNCH 900, NTME,NTME,NTME,NTME,NTME                             BD06E264
        PUNCH 901, (QO(I) , I=1,NTME)                                   BD06E265
        PUNCH 902, (TSTOR(I),I=1,NTME)                                  BD06E266
        PUNCH 903, (PO(I)    ,I=1,NTME)                                 BD06E267
        PUNCH 904, (TSTOR(I),I=1,NTME)                                  BD06E268
        PUNCH 905, (HSTAB(I),I=1,NTME)                                  BD06E269
        PUNCH 906, (TSTOR(I) ,I=1,NTME)                                 BD06E270
        PUNCH 907, (HLTAB(I) ,I=1,NTME)                                 BD06E271
        PUNCH 908, (TSTOR(I) ,I=1,NTME)                                 BD06E272
        PUNCH 909, (UETAB(I) ,I=1,NTME)                                 BD06E273
        PUNCH 910, (TSTOR(I) ,I=1,NTME)                                 BD06E274
    900 FORMAT (13H NTMP=1,NTME= I2, 6H, NPC=,I2, 6H, NHS=,I2,6H, NUF= I2,BD06E275
      1     8H, NHLOC= I2, 1H,  )                                       BD06E276
    901 FORMAT (8H QO    = E18.8,2H, ,E18.8,2H, ,E18.8,1H,  /           BD06E277
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E278
    902 FORMAT (8H TQO   = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E279
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E280
    903 FORMAT (8H PO    = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E281
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E282
    904 FORMAT (8H TPO   = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E283
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E284
    905 FORMAT (8H HSTAB = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E285
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E286
    906 FORMAT (8H THS   = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E287
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E288
    907 FORMAT (8H HLTAB = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E289
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E290
    908 FORMAT (8H THLOC = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E291
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E292
    909 FORMAT (8H UETAB = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E293
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E294
    910 FORMAT (8H TUE   = E18.8,2H, ,E18.8,2H, ,E18.8,1H, /            BD06E295
      1( E18.8,2H,  ,E18.8,2H, ,E18.8,2H, ,E18.8,1H, ) )                BD06E296
        WRITE(6,920) NTME,(TSTOP(I),I=1,NTME)                           BD06E297
    920 FORMAT (1H-  I6,31H VALUES PUNCHED, AT TIMES BELOW /(5E16.8) )  BD06E298
        WRITE (6,921)                                                   BD06E299
    921 FORMAT (13H-NOMENCLATURE/  60H QO=QSUBI,  PO=PS,      HSTAB=RECI, BD06E300
      1HLTAB=SHKI, UETAB=VSHK  //  )                                    BD06E301
        WRITE (6,901) ( QO(I),I=1,NTME )                                BD06E302
```

```
      WRITE (6,903) ( PO(I)   ,I=1,NTME)                         BD06E303
      WRITE (6,905) (HSTAB(I),I=1,NTME)                          BD06E304
      WRITE (6,907) (HLTAB(I),I=1,NTME)                          BD06E305
      WRITE (6,909) (UETAB(I),I=1,NTME)                          BD06E306
  804 IF (NDBUG.LE.0) GO TO 810                                  BD06E307
  805 CALL OUTPUT                                                BD06E308
  810 WRITE (6,811)                                              BD06E309
  811 FORMAT ( 1H1 )                                             BD06E310
      ALT = 0.0                                                  BD06E311
      IGO = 0                                                    BD06E312
      IF (NOSH.GT.0) TWALL = 0.                                  BD06E313
      IF (MTRANS.LE.0)  GO TO 815                                BD06E314
      TRANR =0.                                                  BD06E315
      TRANLT=0.                                                  BD06E316
      GO TO 110                                                  BU06E317
  815 IF (TRANLT.LE.0.) GO TO 110                                BD06E318
      TRANR =0.                                                  BD06E319
      GO TO 110                                                  BD06E320
C           NEGATIVE TEMPERATURE CHECK  (EACH CALCULATION)       BD06E321
C 850 IF (NDIM.NE.2) GO TO 854
C           NOCHEK = 0 , FOR CHECKING NEG. TEMP., AND STABILITY
C           NOCHEK = 1 , FOR NEGATIVE TEMPERATURE CHECK ONLY
C           , BUT STILL CHECKS FOR 2-D STABILITY  (IN TWOD)
C           NOCHEK = 2 , FOR NO CHECKS ( FASTER RUNNING, BUT ERROR PRONE)
  850 IF (NOCHEK.GT.1)   GO TO 728
      INDX2D=0
      IF (NDIM.NE.2) GO TO 854
  851 INDX2D= INDX2D + 1
      IF (INDX2D.GT.NBCOL) GO TO 2160
      ELB = BB(INDX2D)
      DO 852 NA =1,NAROW
      TEMP(NA) = T2(NA,INDX2D)
C     RHO(NA)  = RHO2(NA,INDX2D)
      CP(NA) = CP2(NA,INDX2D)
      EK(NA) = DK(NA,INDX2D)
  852 EL(NA) = AMIN1(AA(NA),ELB)
      DELBRT = 0.
  854 DELTA1 = DELTA
      DO 855 I=NSEG2,NSEG                                        BD06E323
      IF (TEMP(I)) 860,855,855                                   BD06E324
  855 CONTINUE                                                   BD06E325
      IF (NOCHEK.GT.0)   GO TO 728
      NOT = 0
      IF (NT.LE.1) DELBRT=0
C           IF COLD WALL INPUTS, CHECK 'DELTA' ONLY AT PRINT TIME     724
      IF (INPCW.LE.0)      GO TO 705                                  725
```

```
      IF (DELBRT.GT.0.)   GO TO 728                                          726
C          STABILITY CHECK   OR IS THAT CZEK                           BD06E326
C     DEL IS A STABLE DELTA                                            BD06E327
C     LOGDEL = EXPONENT (10) OF DEL                                    BD06E328
C     CAY =  SCALED (BY CON, FOR EASE OF PRINTOUT) DELTA               BD06E329
C     NRI =  NO. OF  TIMES THROUGH FOR PRINT                           BD06E330
C     WRI =  SAME AS NRI ,    WRI*CAY SHOULD = WRITE                   BD06E331
  705 IF (INDX2D.LE.1)   DEL = DELTA*2.
      IF (KFUNT.NE.4) GO TO 706                                        BD06E333
      IF (DELBRT .GT. 0.) GO TO 728                                    BD06E334
  706 DO 710 IRA = NSEG2,NSEG                                          BD06E335
C                        ONLY IF 2D ANC NO CONDUCTION  - 709
      IF (NDIM.EQ.1) GO TO 704
      IF (CC(IRA,INDX2D).LE.0) GO TO 709
C                        CHECKED ONLY FOR 2-DIM
  704 DELBRT =SCON *RHO(IRA)*CP(IRA)*EL(IRA)**2 / EK(IRA)              BD06E336
      DEL = AMIN1(DEL,DELBRT)                                          BD06E337
      DALETH = DELATH*TEMP(IRA)
      IF (DEL.GT.DALETH)   GO TO 709
      WRITE (6,707) IRA,RHO(IRA),CP(IRA),EL(IRA),EK(IRA),DELBRT,INDX2D  BD06E339
  707 FORMAT (25H-POSSIBLE ERROR, SEGMENT   I3,24H RHO, CP, EL, EK, DELBBD06E340
     FRT   /5E16.8 ,' COLUMN',I6)
      IDELER = IDELER + 1                                              BD06E342
      IF (IDELER.GT.50)  GO TO 870
      DEL = DALETH
      WRITE (6,708) IRA,DEL,DELBRT,TEMP(IRA)
  708 FORMAT (' FOR SEGMENT',I6,', MINIMUM DELTA IS',E13.5,', MAX=',E13.
     15,' AT TEMP = ',E13.5)
      IF (DELTA.LE.0.) DEL = DELBRT
      IF ( RHO(IRA)*CP(IRA)*EL(IRA)*EK(IRA).EQ.0.) GO TO 880
  709 IF (TEMP(IRA).EQ.TEMP(IRA+1)) GO TO 715                          BD06E345
  710 CONTINUE                                                         BD06E346
      IF (NDIM.EQ.2)   GO TO 851
  715 LOGDEL =ALOG10(DEL)                                              BD06E347
      IRA = 1                                                          BD06E348
      CON = 1.0                                                        BD06E349
  716 CAY = CON*10.**LOGDEL                                            BD06E350
      IF (WRITE.LE.0.)  WRITE=AMAX1(.5,DELBRT)
      NRI =  WRITE/CAY                                                 BD06E351
      IF(NRI.LT.1) NRI = 1                                             BD06E352
      IF (CAY.LE.DEL) GO TO 725                                        BD06E353
      IRA = IRA + 1                                                    BD06E354
      GO TO (717,718,719,720,727) ,IRA                                 BD06E355
  727 IF (NRI.GT.0) GO TO 725                                          BD06E356
      GO TO 867                                                        BD06E357
  717 CON = .5                                                         BD06E358
```

```
      GO TO  716                                                        BD06E359
  718 CON = .25                                                         BD06E360
      GO TO 716                                                         BD06E361
  719 CON = .20                                                         BD06E362
      GO TO 716                                                         BD06E363
  720 CON = .10                                                         BD06E364
      GO TO  716                                                        BD06E365
  725 WRI = NRI                                                         BD06E366
      DELTA = WRITE/WRI                                                 BD06E367
C             SHOULD BE APPROX. SAME AS CAY                             BD06E368
      IF (NDIM.EQ.2)  GO TO 851
 2160 IF (DELTA.EQ.DELTA1) GO TO 728
  730 WRITE (6,726) DEL,CON,CAY,WRI,DELTA,LOGDEL                        BD06E370
  726 FORMAT ( 47H STABILITY  CHECK,  DEL, CON, CAY, WRI, DELTA  /      BD06E371
     1  5F20.8,I8 )                                                     BD06E372
C             END   STABILITY   CHECK                                   BD06E373
  729 IF (IGO .GT. 0) GO TO 868                                         BD06E374
      GO TO 400                                                         BD06E375
  860 IGO = 1                                                           BD06E376
      WRITE(6,866)                                                      BD06E377
  866 FORMAT (32H-NEGATIVE TEMPERATURE, CUT DELTA )                     BD06E378
      NOT = NOT + 1                                                     BD06E379
C     CALL PRINT                                                        BD06E380
      TRANSF = 1.0                                                      BD06E380
      IF ((NOT.GT.5).AND.(DELBRT.LT.DEL))   IGO = 0
      IF ( NOT.GE.5 )   GO TO 868
      DELTA = DELTA / 4.                                                BD06E381
      GO TO 705                                                         BD06E382
  867 WRITE (6,861) CAY                                                 BD06E383
  861 FORMAT ( 1H-E20.8,30H DELTA WAS TRIED, EXAMINE DATA  )            BD06E384
      GO TO 870                                                         BD06E385
  868 TIME = TIME1(1)                                                   BD06E386
      ALT = 0                                                           BD06E387
      GO TO 110                                                         BD06E388
  880 WRITE (6,881)
  881 FORMAT ('- CHECK YOUR MATERIAL PROPERTYS ')
  870 NOBUG =1                                                          BD06E389
      CALL PRINT                                                        BD06E390
      WRITE (6,8888)                                                    BD06E392
 8888 FORMAT (45H-UNABLE TO CALCULATE DELTA FROM INITIAL DATA   )       BD06E393
      STOP                                                              BD06E395
      END                                                               BD06 396
```

```
      SUBROUTINE ABLATE                                                 ABLA   2
CABLATE                                                                 ABLA   1
C - - -     MARCH 27 1964     JOHN KOHLENBERGER        FORTRAN IV       ABLA   3
C     SOME COMMENTS - NSEG1= 1ST NON-ABLATING SEGMENT.   NSEG2= 1ST     ABLA   4
C     SEGMENT PRESENT. HENCE NSEG2 LESS THAN OR EQUAL TO NSEG1.         ABLA   5
C     SET IGO = NONZERO TO INDICATE ONLY 1 SEGMENT LEFT.                ABLA   6
C     LEO BERGERS CHANGES TO ABLATE                                     ABLA   7
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)ABLA   8
     1  ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)     ABLA   9
     2  ,   QI(100),QAD1(100), CP(40),EK(40), EL(40),RHO(40),TEMP(40)   ABLA  10
     3  ,   A(4,40),B(4,40)                                             ABLA  11
     4  ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)      ABLA  12
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI ABLA  13
     1  ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                  ABLA  14
     2  ,   AA, BB, CC,CP2, DK, T2                                      ABLA  15
      COMMON       ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV             ABLA  16
     1  ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                        ABLA  17
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,           KASE,KFUNT,    NPQABLA  18
     3  ,   NBCOL,NDBUG,NDIM,NOSB,NPGAS,NCI,NPREST,NPU,NSEG,NSEG1,NSEG2 ABLA  19
     4  ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   ABLA  20
     5  ,   RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLABLA  21
     6  ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR    ABLA  22
     7  ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE        ABLA  23
     8  ,   XX,XDIST                                                    ABLA  24
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                       ABLA  25
      EQUIVALENCE  (NSEG,NAROW)                                         ABLA  26
      IF (NSEG.LT.NSEG2) GO TO 60                                       ABLA  27
      IF (WAB.LE.0.) WABSAV=0.
C  *****    NOTE,   USE QSUBI IN PLACE OF QNET WHEN NPU POSITIVE  ***   ABLA  63
      IF (NOI.GE.1) GO TO 29                                            ABLA
      QUSE = QNET                                                       ABLA
      GO TO 3                                                           ABLA
   29 QUSE = QSUBI                                                      ABL1  67
      IF (QUSE.EQ.0.)   GO TO 3
      QUSE = QUSE * QCONS
    3 DQSAV =(TEMP(NSEG1)-TEMP(NSEG2))*DELTA/(150.*(EL(NSEG2)/EK(NSEG2) ABLA
     1 +EL(NSEG1)/EK(NSEG1)))                                           ABLA
      IF (QUSE.NE.0.)   DQSAV = QUSE *DELTA/3600.
      IF (DQSAV.LE.0.) GO TO 15
    2 IF(NPGAS.GE.0) GO TO 6                                            ABLA  29
      IF (NPQ.LE.0) GO TO 6
  100 IF (STAGI-VAPRES(1))107,105,105                                   ABLA  30
  105 HOABL = FINTRP (VAPRES, TEMPW,STAGI,NPQ,1)                        ABLA  31
```

```
      ISNANA =105                                                       ABLA  32
      GO TO 10                                                          ABLA  33
  107 STAGI = VAPRES(1)                                                 ABLA  34
      GO TO 10                                                          ABLA  35
    6 IF (PSTMAX.LE.0.) GO TO 1                                         ABLA  36
      AMF2 = AMINF**2                                                   ABLA  37
      IF (AMINF .GT. 1.09) GO TO 110                                    ABLA  38
      PSTAG = (PINF+RHOINF*VINF*VINF*.5)/2116.                          ABLA  39
      GO TO 111                                                         ABLA  40
  110 PSTAG=PINF*(1.2*AMF2)**3.5*(6./(7.*AMF2-1.))**2.5/2116.           ABLA  41
  111 IF (PSTAG.LE.PSTMAX) GO TO 1                                      ABLA  42
      HVT=  HVPSTM                                                      ABLA  43
      IF (IREC.EQ.2) GO TO 5                                            ABLA  44
      ETA = ETALP                                                       ABLA  45
      GO TO 9                                                           ABLA  46
    5 ETA = ETATP                                                       ABLA  47
      GO TO 9                                                           ABLA  48
    1 HVT = HV                                                          ABLA  49
      IF (IREC.EQ.2) GO TO 7                                            ABLA  50
      ETA = ETA1                                                        ABLA  51
      GO TO 9                                                           ABLA  52
    7 ETA = ETA2                                                        ABLA  53
    9 HOABL = HVT + ETA*(RECI-WALLI)
      ISNANA = 9
   10 IF (NDBUG.GE.1)  WRITE (6,11) ISNANA,HOABL,HV,HVT,HVPSTM,ETA,ETA1,ETA2,STA
     1 ETA2,STAGI
   11 FORMAT (' IN ABLATE , FROM' ,I6/(6E16.8))                         ABLA  54
      IF (TEMP(NSEG2)-TABL)15,30,20                                     ABLA  55
   15 NSEG1 = NSEG2                                                     ABLA  56
      WABDOT = 0.0                                                      ABLA  57
      IF ((NSEG2.EQ.1).AND.(NSEG.EQ.1)) WABSAV=0.0                      ABLA  58
      GO TO 45                                                          ABLA  59
   20 TEMP(NSEG2) = TABL                                                ABLA  60
   30 NSEG1 = NSEG2+1                                                   ABLA  61
   33 IF (NSEG2.GT.NSEG)GO TO 60                                        ABLA  62
      IF (NSEG1.GT.40) GO TO 60
      IF (NSEG.EQ.1) GO TO 90
   35 DQSAV =(TEMP(NSEG1)-TEMP(NSEG2))*DELTA/(150.*(EL(NSEG2)/EK(NSEG2) ABLA  68
     1 +EL(NSEG1)/EK(NSEG1))) + QUSE*DELTA/3600.                        ABLA  69
      QSAV = QSAV+DQSAV                                                 ABLA  70
      WABDOT = DQSAV/HOABL                                              ABLA  71
      IF (WABDOT) 39,41,41                                              ABLA  72
   39 WABDOT = 0.0                                                      ABLA  73
   41 WAB = WAB+WABDOT                                                  ABLA  74
      WABSAV = WABSAV + WABDOT                                          ABLA  75
      FSEG  =  WABSAV/RHO(NSEG2)                                        ABLA  76
```

```
      FDL   =  FSEG-FL(NSEG2)/12.                                      ABLA  77
      IF (NOBUG.EQ.1) WRITE (6,44) DELTA,QUSF,WABSAV,FSEG,FDL,EL(NSEG2) ABLA A78
     1 , QSAV,DQSAV                                                     ABLA B78
   44 FORMAT (' DEBUG ABLATE *** , DELTA QUSE WABSAV FSEG FDL EL/ Q DQ 'ABLA C78
     1  / (6F16.8))                                                     ABLA D78
      IF (FDL.GE.0.) GO TO 80                                           ABLA  79
   45 CALL ONED                                                         ABLA  80
   55 RETURN                                                            ABLA  81
   60 CALL PRINT                                                        ABLA  82
   65 WRITE  (6,66)                                                     ABLA  83
   66 FORMAT( 77HO IN ABLATION ROUTINE HAVE EXHAUSTED ALL BUT 1 SEGMENT ABLA  84
     1SO AM STOPPING.             )                                     ABLA  85
C     PROBABLY PRINT SOME OTHER STUFF HERE TOO                          ABLA  86
      IGO = 1                                                           ABLA  87
      GO TO 55                                                          ABLA  88
   80 WRITE (6,81) NSEG2                                                ABLA  89
   81 FORMAT( 42HO  HAVE COMPLETED ABLATION ON SEGMENT NO.   I3 /55H  NOABLA  90
     1TE THAT ON OUTPUT THERE IS NOW 1 LESS TEMPERATURE.              )ABLA  91
      WABSAV=  RHO(NSEG2)*FDL                                           ABLA  92
      NSEG2 = NSEG2+1                                                   ABLA  93
      TEMP(NSEG2) = TABL                                                ABLA  94
      NSEG1 = NSEG2 + 1                                                 ABLA  95
      IF (NSEG2.GT.40) GO TO 60                                         ABLA  96
      IF (NSEG1.GT.NSEG) GO TO 60
   83 CALL PRINT                                                        ABLA  97
   85 QSAV = 0.0                                                        ABLA  98
      GO TO 45                                                          ABLA  99
   90 WRITE (6,91)
   91 FORMAT ('1 ADD MORE SEGMENTS '/ ' WHEN ABLATING, MORE THAN ONE SEG
     1MENT IS REQUIRED ')
      GO TO 60
      END                                                               ABLA 100
```

```
      SUBROUTINE ABLUNT                                                 BLUNT 10
CABLUNT  AEROSPACE NOSE BLUNTING ROUTINE (B006C PROGRAM)                 BLUNT  1
C     ENGINEER GEYTVAY, PROGRAMMER KOHLENBERGER  02/07/64                BLUNT  2
C - - -     APRIL 14, 1964    JOHN KOHLENBERGER    ** FORTRAN II **       BLUNT  3
C           BASED ON STAGNATION POINT RECESSION AND HANKEY TYPE HEATING BLUNT  4
C            AT THE SPHERE JUNCTION.                        4-14-64     BLUNT  5
C               JULY  5 1967                                             BLUNT  6
C     SOME COMMENTS - NSEG1 = FIRST NON-ABLATING SEGMENT                 BLUNT  7
C                     NSEG2 = FIRST SEGMENT PRESENT                      BLUNT  8
C                                                                        BLUNT  9
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)BLUNT 11
     1  ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)      BLUNT 12
     2  ,   QI(100),QAD1(100), CP(40),EK(40),FL(40),RHO(40),TEMP(40)     BLUNT 13
     3  ,   A(4,40),B(4,40)                                              BLUNT 14
     4  ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)       BLUNT 15
      COMMON      A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI BLUNT 16
     1  ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                   BLUNT 17
     2  ,   AA, BB, CC,CP2, DK, T2                                       BLUNT 18
      COMMON      ALT,AMINF,AOFA,ANGLE, CONPR, DELTA, DQSAV              BLUNT 19
     1  ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                         BLUNT 20
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IRFC,          KASE,KFUNT,    NPQBLUNT 21
     3  ,   NBCOL,NOBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 BLUNT 22
     4  ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   BLUNT 23
     5  ,   RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLBLUNT 24
     6  ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR    BLUNT 25
     7  ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE         BLUNT 26
     8  ,   XX,XDIST                                                     BLUNT 27
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                       BLUNT 28
      COMMON / SWALOW/  QINT(50),HEDGE(50),PEDGE(50),UEDGE(50),HRECOV(50BLUNT 29
     1)  ,  INPCW ,ISET                                                  BLUNT 30
      COMMON / MIKE / HV2,ETAL2,ETAT2,XI, FV1P(10),FV1S(10),FV2P(10),    BLUNT 31
     *      FV2S(10),STGTBL(10),SHRTBL(10),NOPSTG,NOPSHR                 BLUNT 32
     2  ,   QSONIC,SHEAR,HOABLM,HF, ISTETS                               BLUNT 33
      DIMENSION   THETAC(11),OJOQS(11)                                   BLUNT 34
      DATA          FV1/1.0/,FV2/0./                                     BLUNT 35
      DATA COSPHI /.74/                                                  BLUNT 36
      DATA  THETAC/  0.0,2.5, 5.,7.5,10.,12.5,15.,20.,25.,30.,35./       BLUNT 37
     1      ,QJOQS /    0.,.004,.016,.035,.057,.085,.115,.18,.255,.335,  BLUNT 38
     2      .42   /                                                      BLUNT 39
      EQUIVALENCE (NSEG,NAROW)                                           BLUNT 40
      AMINF = VINF/(49.07*SQRT (TINF))                                   BLUNT 41
      A1 = .884*AMINF**(-1.768)                                          BLUNT 42
      RHOSL = 0.0023769                                                  BLUNT 43
```

```
C              AS PER ECKERT                                          BLUNT 44
C     FROM EQU 25 , PG. 7  (AE 62-0497)                               BLUNT 45
      QDSRD = 17600.*SQRT (RHOINF/RHOSL)* (VINF/26000.)**3.15         BLUNT 46
     1      * (STAGI-WALLI)/(STAGI-129.5)                              BLUNT 47
      QDOT = QDSRD / SQRT (RADIUS)                                     BLUNT 48
      HSUBT = QDSRD*3600./(SQRT (RADIUS)*(RECI-WALLI))                 BLUNT 49
      AMF2 = AMINF**2                                                  BLUNT 50
    9 IF (PSTMAX.LE.0.) GO TO 13                                       BLUNT 51
      IF (AMINF.GT.1.09)  GO TO 1                                      BLUNT 52
      PSTAG =(PINF + RHOINF*VINF**2 *.5)/2115.8                        BLUNT 53
      GO TO 11                                                         BLUNT 54
    1 PSTAG = PINF* (1.2*AMF2)**3.5 *(6./(7.*AMF2-1.))**2.5 / 2116.    BLUNT 55
   11 IF (PSTAG.LE.PSTMAX) GO TO 13                                    BLUNT 56
      HVT = HVPSTM                                                     BLUNT 57
      ETA = ETALP                                                      BLUNT 58
      GO TO 14                                                         BLUNT 59
   13 HVT = HV                                                         BLUNT 60
      ETA = ETA1                                                       BLUNT 61
      IF (AMINF.GT.1.09) GO TO 12                                      BLUNT 62
      PSTAG =(PINF + RHOINF*VINF**2 *.5)/2115.8                        BLUNT 63
      GO TO 14                                                         BLUNT 64
   12 PSTAG = PINF* (1.2*AMF2)**3.5 *(6./(7.*AMF2-1.))**2.5 / 2116.    BLUNT 65
   14 XIM1 = 1.-XI                                                     BLUNT 66
      IF (INB4.LE.0)  FV1=1.0
      IF (INB4.LE.0)  FV2=0.0
      IF (XI.EQ.1.0)  GO TO 16
      IF (NOPSTG.LE.0)  GO TO 155
      FV1 = FINTRP(STGTBL,FV1P,PSTAG,NOPSTG,1)                         BLUNT 68
      FV2 = FINTRP(STGTBL,FV2P,PSTAG,NOPSTG,1)                         BLUNT 69
  155 ETA = ETA1                                                       BLUNT 70
   16 DELTAI = RECI-WALLI                                              BLUNT 71
      XIHV1 = XI *(HVT +ETA *DELTAI)                                   BLUNT 72
      XIHV2 = XIM1*(HV2+ETAL2*DELTAI)                                  BLUNT 73
      HFDI = XI*(HF+ETA*DELTAI)                                        BLUNT 74
      HOABL = FV1 * XIHV1  + FV2 * XIHV2 + (1.-FV1) * HFDI             BLUNT 75
      IF (NDBUG.GE.1)
     1WRITE (6,14922)   HOABL,FV1,XIHV1,FV2,XIHV2,HFDI,HVT,ETA,XI,DELTAI
14922 FORMAT (' FROM ABLUNT AT FN 14922'/(5E16.8))
C              HOABL AT STAG POINT                                     BLUNT 76
      IF(INB4-1) 8,8,10                                                BLUNT 77
C              INITIALIZE CONSTANTS AT 8  (BELOW)                      BLUNT 78
    8 INB4 = 2                                                         BLUNT 79
      F = 0.0                                                          BLUNT 80
      FSAV = 0.0                                                       BLUNT 81
      IF (ANGLE-10.0) 7,7,6                                            BLUNT 82
    7 A2 = 1.00                                                        BLUNT 83
```

```
      GO TO 5                                                           BLUNT 84
    6 A2 = .70*(COS (.03925*(ANGLE-10.0)))**3.74 + .30                  BLUNT 85
    5 RO = RADIUS                                                       BLUNT 86
      F = 0.0                                                           BLUNT 87
      THETA = ANGLF / 57.2957795                                        BLUNT 88
      SITHET = SIN (THFTA)                                              BLUNT 89
      IF (ANGLE-20.0)  3,3,4                                            BLUNT 90
    4 B2 = SITHET**1.50                                                 BLUNT 91
      GO TO 2                                                           BLUNT 92
    3 B2 = 1.480*SITHET**1.943                                          BLUNT 93
    2 B3 = 1.047*COS (THETA)/SITHET                                     BLUNT 94
      F = 0.0                                                           BLUNT 95
      Z = 0.0                                                           BLUNT 96
      THMACH = 90. - ANGLE                                              BLUNT 97
      ANGLP = THMACH/57.2957795                                         BLUNT 98
      AONE  = 1.C47* SIN (ANGLR)/COS (ANGLR) -2.095                     BLUNT 99
   10 IF (QNET) 19,19,15                                                BLUNT100
   15 IF (TEMP(NSEG2)-TWNB)20,45,30                                     BLUNT101
   30 TEMP(NSFG2) = TWNB                                                BLUNT102
C            4-14-64      BASIC EQUNS. BELOW                            BLUNT103
   45 DF = QDSPD * DELTA / (SQRT (RADIUS)*RHO(NSEG2)*HOABL)             BLUNT104
      F   = F + DF                                                      BLUNT105
      NSEG1 = NSFG2 + 1                                                 BLUNT106
      B1 = (SITHET - B2 - A1*A2) / (1. - SITHET)                        BLUNT107
      PR = .76                                                          BLUNT108
      IF (( ISTETS.GT.0) .AND. (IREC.NE.2))  GO TO 100                  BLUNT109
      FACTOR = .637                                                     BLUNT110
      IF (IREC.EQ.2) FACTOR = 4.68*(RHOINF/RHOSL)**.3 *RADIUS**.3       BLUNT111
C             SONIC POINT JAZZ -REPLACES(FDF=FACTOR*DF)                 BLUNT112
      QSONIC = QDSRD/SQRT(RADIUS) * FACTOR                              BLUNT113
      SHEAR  =  QSONIC*VSHK/ (DELTAI* PR**(-.667))/32.2                 BLUNT114
      IF (XI.EQ.1.)     GO TO 449                                       BLUNT115
          PS = PSHK/2116.                                               BLUNT116
      IF (NOPSTG.LE.0)  GO TO 446
      FV1 = FINTRP(STGTBL,FV1P,PS   ,NOPSTG,1)                          BLUNT117
      FV2 = FINTPP(STGTBL,FV2P,PS   ,NOPSTG,1)                          BLUNT118
  446 IF (NOPSHR.LE.0)  GO TO 447
      FV1SHR = FINTRP(SHRT3L,FV1S,SHEAR,NCPSHR,1)                       BLUNT119
      FV2SHR = FINTRP(SHRTBL,FV2S,SHEAR,NOPSHR,1)                       BLUNT120
  447 IF (IREC.NF.2) GO TO 448                                          BLUNT121
      XIHV1 = XI   *(HVT +ETA2 *DELTAI)                                 BLUNT122
      XIHV2 = XIM1*(HV2 +ETAT2*DELTAI)                                  BLUNT123
      HFDI = XI*(HF+ETA2*DELTAI)                                        BLUNT124
  449 HOABLP= FV1   * XIHV1+  FV2 * XIHV2  + HFDI * (1.-FV1)            BLUNT125
CHANGED     9.29.69
      IF (NOPSHR .GT. 0)   GO TO 443
```

```
      HOABLM = HOABLP
      GO TO 444
  443 HOABLS = FV1SHR*XIHV1 + FV2SHR*XIHV2 + HFDI * (1.-FV1SHR)         BLUNT126
      HOABLM = AMIN1(HUABLP,HOABLS)                                     BLUNT127
  444 FDF =  QSONIC/HOABLM*DELTA/RHO(NSEG2)                             BLUNT128
CHANGED    9.29.69
      GO TO 450                                                         BLUNT129
  449 FDF = FACTOR * DF                                                 BLUNT130
      HOABLM = HOABL
 450  ISET = 0
      IF (NDBUG.GE.1)   WRITE(6,451)  PS,SHEAR,FV1,FV2,FV1SHR,FV2SHR,
     1 XIHV1,XIHV2,HFDI,HOABLP,HOABLS,HOABLM,    FDF,IREC,NOPSTG,NOPSHR
  451 FORMAT (' IN ABLUNT, AT 450   PS,SHEAR,FV1-2,FV1-2SHR '/6E16.8/
     1 ' XIHV1-2,HFDI,HOABLP-S-M '/ 6E16.8/ ' FDF,IREC,NOPSTG-SHR '/
     2 E16.8,3I6 )
      IF ( (AMAX1(DF,FDF)).LT.((.999-COSPHI)*RADIUS))  GO TO 36         BLUNT132
      ISET = 1                                                          BLUNT133
      DELTA = DELTA / 2.                                                BLUNT134
      WRITE (6,35)  DF,FDF,RADIUS,DELTA                                 BLUNT135
   35 FORMAT ( 54H- DF OR FDF .GT. RADIUS*.05,   DF,FDF,RADIUS,DELTA    BLUNT136
     *  /  4F16.8)                                                      BLUNT137
      WRITE(6,6666)IREC,RN,FDFACC,DFFAC,FDFAC,FDF,HOABLS,               BLUNT138
     *       HOABLP,FACTOR,XIHV1,XIHV2,FV1,FV2 ,PSTAG,PSHK,SHEAR,QSONIC BLUNT139
      IF (DELTA.LT. .1E-8) STOP                                         BLUNT140
      RETURN                                                            BLUNT141
C            ERROR RETURN       DELTA HAS BEEN CUT                      BLUNT142
   36 DFFAC = 1.-DF/RADIUS                                              BLUNT143
      FDFAC = 1.-FDF/RADIUS                                             BLUNT144
      FDFACC =      FDFAC *COSPHI                                       BLUNT145
      RN =  RADIUS * (FDFAC**2 -2.*DFFAC*FDFACC +DFFAC**2) *.5 /        BLUNT146
     *      (-FDFACC+DFFAC)                                             BLUNT147
 1776 IF (NDBUG.NE.2) GO TO 1967                                        BLUNT148
      WRITE(6,6666)IREC,RN,FDFACC,DFFAC,FDFAC,FDF,HOABLS,               BLUNT149
     *       HOABLP,FACTOR,XIHV1,XIHV2,FV1,FV2 ,PSTAG,PSHK,SHEAR,QSONIC BLUNT150
 6666 FORMAT ( 24H BLUNT,DEBUG=2,  IREC = I3/34H RN, FDFACC, DFFAC, FDFABLUNT151
     *C, HOABLS   /6E16.8/ 31H HOABLP, FACTOR, XIHV1,2, FV1,2 /(6E16.8))BLUNT152
 1967 B4 = B2 + A1*A2                                                   BLUNT153
      DDR = B4*DF                                                       BLUNT154
      VA = B3*(RN+DDR)**3-2.095*RN**3-RADIUS*RADIUS*RADIUS*AONE         BLUNT155
      RADIUS = RN                                                       BLUNT156
      VTOTAL = VTOTAL + VA                                              BLUNT157
      WABDOT = DF*RHO(NSEG2)                                            BLUNT158
      WAB = WAB + WABDOT                                                BLUNT159
      VAPRES(100) = F                                                   BLUNT160
      FSAV = FSAV +DF                                                   BLUNT161
      FDL=FSAV-EL(NSEG2)/12.                                            BLUNT162
```

```
      IF (FDL) 20,40,40                                                 BLUNT163
   40 FSAV = FDL                                                        BLUNT164
      NSEG2 = NSEG2 +1                                                  BLUNT165
      NSEG1 = NSEG2 +1                                                  BLUNT166
      TEMP(NSEG2) = TWNB                                                BLUNT167
      WRITE (6,81) NSEG2                                                BLUNT168
   81 FORMAT (42HO  HAVE COMPLETED ABLATION OF SEGMENT NO.  I3 / 54H CONBLUNT169
     1DITIONS JUST PREVIOUS TO ABLATICN ARE AS FOLLOWS  )               BLUNT170
      IF (NSEG1.GT.NSEG) GO TO 90
      CALL PRINT                                                        BLUNT172
      GO TO 20                                                          BLUNT173
   19 NSEG1 = NSEG2                                                     BLUNT174
   20 CALL ONED                                                         BLUNT175
   48 IF (NDBUG-2)  50,46,50                                            BLUNT176
C             NOTE/  DQSAV AND QSAV NOT USED                            BLUNT177
   46 DQSAV = DDR                                                       BLUNT178
      QSAV  = VA                                                        BLUNT179
C             NOTE/  SO DEBUG WILL PRINT DDR AND VA                     BLUNT180
      WRITE             (6,47)ODSRD,DQSAV,QSAV,WABDOT,ETA,RADIUS,ANGLR  BLUNT181
     6  ,VTOTAL,DF,F,  WAB,A1,A2,B2,B4,  AMINF                          BLUNT182
   47 FORMAT ( 7HOAPLUNT/7X,6H QDSRD 15X,5HDQSAV 15X,4HQSAV 16X,        BLUNT183
     1  6HWABDOT 14X,3HETA / 5F20.8 //7X,7H RADIUS 14X,5HANGLR          BLUNT184
     2  15X,6HVTOTAL 14X,2HDF 18X,1HF / 5F20.8 /7X,4H WAB 16X,2HA1 18X, BLUNT185
     3  2HA2 18X,2HB2 18X,2HB4 / 5E20.8 / 7X,6H AMINF /E20.8  )         BLUNT186
   50 IF (RADIUS) 60,60,80                                              BLUNT187
   80 IF (RADIUS - 1.E5) 55,60,60                                       BLUNT188
   55 RETURN                                                            BLUNT189
   60 CALL PRINT                                                        BLUNT190
   65 WRITE (6,66)                                                      BLUNT191
   66 FORMAT (77 HONOSE BLUNTING ROUTINE HAS NEGATIVE OR LARGE RADIUS,  BLUNT192
     1SO AM STOPPING.                    )                              BLUNT193
C     PROBABLY PRINT SOME OTHER STUFF HERE TOO                          BLUNT194
   90 IGO = 1                                                           BLUNT195
      GO TO 55                                                          BLUNT196
C               ----- STETSON -----                                     BLUNT197
  100 QTANQS = FINTRP (THETAC,QJOQS,ANGLE, 11,1)                        BLUNT198
      QTAN   = QTANQS * QDOT                                            BLUNT199
      QSONIC = QTAN                                                     BLUNT200
      PTOPS =  SIN(ANGLE)**2 + COS(ANGLE)**2/(1.2*AMINF)                BLUNT201
      PTAN   =  PTOPS * PSTAG                                           BLUNT202
      VTAN   = VINF*  SQRT (1.-PTOPS**.16667                            BLUNT203
      SHEAR = QTAN*VTAN *PR**(-.6667) / (32.2*DELTAI)                   BLUNT204
C                 REPRODUCED FROM  IFN.78 TO IFN.83, OR THERE ABOUTS    BLUNT205
      IF (XI.EQ.1.) GO TO 101                                           BLUNT206
      IF (NOPSTG.LE.0)  GO TO 102
      FV1   = FINTRP(SIGTBL,FV1P,PTAN,NOPSTG,1)                         BLUNT207
```

```
      FV2  = FINTRP(STGTBL,FV2P,PTAN,NOPSTG,1)                          BLUNT208
  102 IF (NOPSHR.LE.0)  GO TO 101
      FV1SHR = FINTRP (SHRTBL,FV1S,SHEAR,NOPSTG,1)
      FV2SHR = FINTRP (SHRTBL,FV2S,SHEAR,NOPSTG,1)
C             SKIPPED A FEW HERE ,DOWN TO EFN.448                       BLUNT211
  101 HOABLP = FV1    *XIHV1+FV2    *XIHV2 +HFDI*(1.-FV1    )           BLUNT212
CHANGED     9.29.69
      IF (NOPSHR .GT. 0)   GO TO 103
      HOABLM = HOABLP
      GO TO 104
  103 HOABLS = FV1SHR*XIHV1+FV2SHR*XIHV2 +HFDI*(1.-FV1SHR)              BLUNT213
      HOABLM = AMIN1 (HOABLP,HOABLS)                                    BLUNT214
C             END OF DUPED STUFF                                        BLUNT215
  104 RN=0.
CHANGED     9.29.69
      IF (HOABLM.GT.0. )
     1PN =  RADIUS+ DF*( SITHET    /(1.-SITHET    )- 1./(1.-SITHET    ) BLUNT216
     1     *  QTANQS*HOABL/HOABLM)                                      BLUNT217
      GO TO 1776                                                        BLUNT218
      END                                                               BLUNT219
```

```
      SUBROUTINE  AEROSP                                                SBO.   1
      WRITE (6,1960)                                                    SBO.   2
 1960 FORMAT (1H1///25X,42H THE MATHEMATICS AND COMPUTATIONS CENTER     SBO.   3
     1/// 45X,3H OF /// )                                               SBO.   4
      WRITE (6,1961)                                                    SBO.   5
 1961 FORMAT (    45X,1H*/ 38X,1H*,13X,1H*/50X,3H***/49X,4H****/ 30X,1H* SBO.   6
     1 16X, 6H******,7X,1H* / 29X,1H*,16X,7H*** ***,8X,1H* / 44X,3H***, SBO.   7
     2 3X,3H***/ 27X,1H*,15X,3H*** 4X,3H*** 10X,1H*/ 41X,3H*** 3X,1H*   SBO.   8
     3 2X,3H***                             /26X,1H* 13X,3H*** 3X,2H** SBO.   9
     4 2X,3H*** 11X,1H* /38X,3H*** 3X,4H****  2X,3H***/ 37X, 3H*** 3X,  SBO.  10
     5 5H***** 2X,3H*** /25X,1H* 9X,3H*** 3X,7H******* 2X,3H*** 12X,1H*/SBO.  11
     6 34X,3H*** 3X,8H********  2X,3H***/32X,3H*** 15X,3H***/26X,1H*,3X,SBO.  12
     7 3H*** 17X,3H*** 11X,1H*/29X,3H*** 4X,12H************ 2X,3H***  / SBO.  13
     8 27X,4H****,4X,13H************* 2X,3H*** 10X,1H*/28X,2H** 4X,     SBO.  14
     9 14H************** 2X,50H**AEROSPACE CORPORATION, SAN BERNARDINO OSBO.  15
     OPERATIONS     /           29X,1H* 31X,1H*/30X,1H*,29X,1H*///    SBO.  16
     9 38X,1H* 13X,1H* /45X,1H*  )                                      SBO.  17
      WRITE (6,1962)                                                    SBO.  18
 1962 FORMAT (1H- ,35X , 15H PRESENTS ..... //  )                       SBO.  19
      RETURN                                                            SBO.  20
      END                                                               SBO.  21
```

```
      SUBROUTINE ATMOS                                                  ATMS   2
CATMOS                                                                  ATMS   1
C     SUBROUTINE ATMOS (ICK,ALT,TINF,PINF,RHOINF)                       ATMS   3
C - - -   JUNE  19 1969     JOHN KOHLENBERGER          FORTRAN IV       ATMS   4
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),ADFA1(100),HINSD1(100)ATMS   5
     1  ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)    ATMS   6
     2  ,    QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)    ATMS   7
     3  ,    A(4,40),B(4,40)                                            ATMS   8
     4  ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)     ATMS   9
      COMMON       A,ALT1,ADFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI ATMS  10
     1  ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                 ATMS  11
     2  ,    AA, BB, CC,CP2, DK, T2                                     ATMS  12
      COMMON       ALT,AMINF,ADFA,ANGLE, CCNPR, DELTA, DQSAV            ATMS  13
     1  ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                       ATMS  14
     2  ,ICHK,ICK,ICONF,IGO,IN34,IN842,IRFC,           KASE,KFUNT,    NPQATMS  15
     3  ,    NBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 ATMS  16
     4  ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  ATMS  17
     5  ,    RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLATMS  18
     6  ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   ATMS  19
     7  ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE       ATMS  20
     8  ,    XX,XDIST                                                   ATMS  21
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      ATMS  22
      COMMON /WERNER/   TRVRSL,TFB(100),RHOFB(100),PFB(100),ALTCM(100)
      EQUIVALENCE (NSEG,NAROW)                                          ATMS  23
C           ADD
      DIMENSION   HB(7),ALM(6),PB(6),TMB(6)
C *************  1959 A.R.D.C. ATMOSPHERE  ***************
      DATA  HB/0.,11000.,25000.,47000.,53000.,79000.,90000./,
     1 ALM/-.0065,0.0000,.00300,0.0000,-.0045,0.0/,  RHOSEL/.0023796/,
     2 PB /760.0,169.76,18.668,.90342,.43749,.007573/,TSL/287.51/,
     3 TMB/288.16,216.66,216.66,282.66,282.66,165.56/,PSEALV/760.0/,
     4 AK/.3048/, C1/.034164794/,RE/6356776./
C *************  1959 A.R.D.C. ATMOSPHERE  ***************
      DATA XEDE/0./
C           ADD
      IF (XEDE.NE.0.)   GO TO 99999
      PSL = PSEALV
      RHOSL = RHOSEL
C           BEGIN REPLACEMENT
99999 ICK = 1
      IF (TRVRSL.GT.0.) GO TO 400
    1 HK = AK*ALT
      RH = RE*HK/(RE+HK)                                                ATMS  30
```

```
C             RE = RADIUS OF EARTH METERS   = 20850000.FT                    ATMS  28
         DO 5  J=1,7
         IF (BH .GE. HB(J))   GO TO 5
C             FOR NEGATIVE ALTITUDES
         IF (J.EQ.1)      GO TO 300
         I = J - 1
         BHB = BH-HB(I)
         GO TO 6
       5 CONTINUE
C             FOR ALTITUDES ABOVE 295278 FT.
         GO TO 300
       6 TMPB = TMB(I)+ALM(I)*BHB
C             TAKES PLACE OF JSWA = 2, AND TEST
         IF ((I/2)*2 .EQ. I      ) GO TO 280
     260 E =C1/ALM(I)
         P = PB(I)* (TMB(I)/TMPB) **E
         GO TO 290
     280 E =-C1* BHB/ TMB(I)
         P = PB(I) * EXP(E)
C              END   REPLACEMENT
     290 DELT = P/PSL
         THETA = TMPB/TSL                                                    ATMS  79
         PINF = 2116.2*DELT                                                  ATMS  80
         TINF = 518.69*THETA                                                 ATMS  81
         RHOINF = (0.0023769*DELT)/THETA                                     ATMS  82
         SIGMA = RHOINF/RHOSL                                                ATMS  83
         IF (NDBUG.EQ.3)   WRITE (6,295) I,BHB,E,P, DELT,THETA,PINF,TINF,    ATMS  84
        1 RHOINF,SIGMA
     295 FORMAT (' DEBUG-ATMOS-, I=',I6,' BHB=',E13.5,' E=',E13.5,' P=',E13
        1.5/ ' DELT, THETA, PINF, TINF, RHOINF, RHOINF/RHOSL'/ 6E13.5)
         GO TO 310
     300 ICK = 0                                                             ATMS  85
         IF ( (XEDE.EQ.0.) .OR. (SIGN(1.,ALT).NE.SIGN(1.,XEDE)) .OR.         ATMS  86
        1    (NDBUG.EQ.3) )  WRITE(6,305) ALT
         XEDE = SIGN(1.,ALT)
     305 FORMAT ('- OUTSIDE ATMOS TABLES, ALTITUDE=' E16.8)
     310 RETURN
     400 NTRAV = TRVRSL                                                      ATMS  87
         SLOPE = 1.
         IF (ALTCM(2).LT.ALTCM(1)) SLOPE = -1.                               ATMS5989
         DO 410 I=1,NTRAV                                                    ATMS5990
         J=I-1                                                               ATMS5991
         IF ( I.GE.2 )SLOPE=1.                                               ATMS5992
         IF ((I.GE.2).AND.(ALTCM(I+1).LT.ALTCM(I)))SLOPE =-1.                ATMS59
         ALTFT = ALTCM(I)/30.48010                                           ATMS59
         IF (SLOPE.GT.0.) GO TO 405                                          ATMS5995
                                                                             ATMS5996
```

```
      IF (ALT.GT.ALTFT) GO TO 415                                        ATMS5997
      GO TO 410                                                          ATMS5998
  405 IF (ALT.LT.ALTFT) GO TO 415                                        ATMS5999
  410 CONTINUE                                                           ATMS-100
      WRITE (6,411) J,ALT,ALTFT,NTRAV,SLOPE                              ATMS-101
  411 FORMAT (' IN-ATMOS-NO F.B. FOUND.  AT END OF LOOP WITH-J,ALT,ALTFTATMS-102
     1,NTRAV' / 16,2E16.8,I6, F16.8)                                     ATMS-103
      GO TO 1                                                            ATMS-104
  415 IF (NOBUG.EQ.3) WRITE (6,416) NTRAV,SLOPE,J,I,ALTCM(J),ALTFT,ALTCMATMS-105
     1(J+1),ALT,TFB(J),TFB(J+1)                                          ATMS-106
  416 FORMAT (' IN-ATMOS,F.B.- NTRAV,SLOPE,J,I' /I5,E16.8,2I6/' ALTCM,ALATMS-107
     1TFT,ALTCM(J+1),ALT,TFB,TFB(J+1)'/ 6E16.8)                          ATMS-108
      IF((TFB(J).LE.0.).OR.(TFB(J+1).LE.0.)) GO TO 1
      FACTOR = (ALT*30.4801-ALTCM(J)) / (ALTCM(J+1)-ALTCM(J))            ATMS  92
      TINF =((TFB(J+1)-  TFB(J)  )* ACTOR + TFB(J))*1.8                  ATMS  93
      RHOINF=((RHOFB(J+1) -RHOFB(J))*FACTOR + RHOFB(J))*1.94057          ATMS  94
      PINF =((PFB(J+1)-  PFB(J)   )*FACTOR + PFB(J))*2.0886E-3
      GO TO 310                                                          ATMS  95
      END                                                                ATMS  97
```

```
      BLOCK DATA                                                        BLK1   1
      COMMON /BLOK1/ TITLE(20)
      DATA TITLE/20*'    '/
      COMMON / MIKE / HV2,ETAL2,ETAT2,XI, FV1P(10),FV1S(10),FV2P(10),   BLK1   4
     *       FV2S(10),STGTBL(10),SHRTBL(10),NOPSTG,NOPSHR               BLK1   5
     2  ,    QSONIC,SHEAR,HOABLM,HF, ISTETS,  NOCHEK,
     3       RHOTBL(10,40),CPTBL(10,40),EKTBL(10,40),TEMTBL(10,40)
      COMMON / SWALOW/  QLIT(50),HEDGE(50),PEDGE(50),UEDGE(50),HRECOV(50BLK1   7
     1)  ,   INPCW ,ISET                                                BLK1   8
C            FOR ARCHIE OSSIN - COLD WALL INPUTS FROM ENTROPY SWALLOWING BLK1   9
      COMMON /CHARAB/  NTMF,PROPTY(300)
      COMMON /WERNER/  TRVRSL,FIRBAL(400)
      COMMON /RALPH/ TWTAB(20),EMTAB(20),NEMIS,NQCOR,QABSF(20),TWA(20)
C     FOR DR. RALPH GILBERT 10/17/70-10/29/70
C    9      ,ETATP,ETALP,HVPSTM,PSTMX,NT                                11/25/70
C
      DATA  FIRBAL/400*0./, RHOTBL/400*0./,CPTBL,EKTBL,TEMTBL/1200*0./
     1 , NTMF/0/,PROPTY/300*0./, QSONIC,SHEAR,HOABLM,HF/4*0./
      DATA XI/1.0/, INPCW/0/  ,ISET/0/,ISTETS/0/ ,NOCHEK/0/            BLK1  10
     1,NOPSTG,NOPSHR/2*0/
      DATA TWTAB,EMTAB,NEMIS,NQCOR,QABSF,TWA /40*0.,2*0,40*0./
      END                                                               BLK1  11
```

```
      SUBROUTINE CONSTQ                                                 CRVQ   3
CCONSTQ  -  CALCULATES QSUBI ON BASIS OF  ALT. OR TIME                  CRVQ   1
C - - -     OCTOBER 14 1970   JOHN KOHLENBERGER         FORTRAN IV      CRVQ   2
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)CRVQ   4
     1    ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)  CRVQ   5
     2    ,    QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40) CRVQ   6
     3    ,    A(4,40),B(4,40)                                          CRVQ   7
     4    ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)   CRVQ   8
C
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI CRVQ   9
     1    ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1               CRVQ  10
     2    ,    AA, BB, CC,CP2, DK, T2                                   CRVQ  11
      COMMON       ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV             CRVQ  12
     1    ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                     CRVQ  13
     2    ,ICHK,ICK,ICONF,IGO,INR4,INR42,IREC,       KASE,KFUNT,     NPQCRVQ  14
     3    ,    NBCOL,NDBUG,NDIM,NOSB,NPGAS,NCI,NPREST,NPU,NSEG,NSEG1,NSEG2 CRVQ  15
     4    ,    NTFAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  CRVQ  16
     5    ,    RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLCRVQ  17
     6    ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR  CRVQ  18
     7    ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE     CRVQ  19
     8    ,    XX,XDIST                                                 CRVQ  20
     9    ,ETATP,ETALP,HVPSTM,PSTMAX                                    CRVQ  21
C
      COMMON /RALPH/ TWTAB(20),EMTAB(20),NEMIS,NQCOR,QABSF(20),TWA(20)
      EQUIVALENCE  (NSEG,NAROW)                                         CRVQ  22
  100 IF (NPQ .GE. 2) GO TO 160                                         CRVQ  23
C          OTHERWISE USE QSOLAR                                         CRVQ  24
  110 QSUBI =  QI(100)                                                  CRVQ  25
C          (QI(100) = 3600  QAD1(100) = QSOLAR )                        CRVQ  26
      GO TO 200                                                         CRVQ  27
  160 IF (TIME-TIME1(1))  7,5,5                                         CRVQ  28
    5 QSUBI = FINTRP (TIME1,QI,TIME,NPGAS,1)                            CRVQ  29
      GO TO 200                                                         CRVQ  30
    7 QSUBI = CPG(1)                                                    CRVQ  31
C          VARIABLE EMIS - ADDED 10/14/70
  200 IF (NEMIS.GT.0) EMIS = FINTRP(TWTAB,EMTAB,TWALL,NEMIS,1)
      IF (NQCOR.GT.0)  QCONS = QCONS * FINTRP(TWA,QABSF,TWALL,NQCOR,1)
      QNET = QSUBI*QCONS - (.174*EMIS* (TWALL/100.)**4)                 CRVQ  34
C          TAKEN FROM -ENERGY-                                          CRVQ  35
      RETURN                                                            CRVQ  36
      END
```

```
      SUBROUTINE DIST                                                   DISTX  3
CDIST                                                                   DISTX  1
C - - -      MARCH 27 1964      JOHN KOHLENBERGER         FORTRAN IV    DISTX  2
C     SUBROUTINE DIST (ICONF,REYN,TRANR,XDIST,XX)                       DISTX  4
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)DISTX  5
     1  ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)    DISTX  6
     2  ,    QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)   DISTX  7
     3  ,    A(4,40),B(4,40)                                            DISTX  8
     4  ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)     DISTX  9
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI DISTX 10
     1  ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                 DISTX 11
     2  ,    AA, BB, CC,CP2, DK, T2                                     DISTX 12
      COMMON       ALT,AMINF,AOFA,ANGLE, CONPR, DFLTA,DQSAV             DISTX 13
     1  ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                       DISTX 14
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,          KASE,KFUNT,    NPQDISTX 15
     3  ,    NBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 DISTX 16
     4  ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  DISTX 17
     5  ,    RADIUS,REFCI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLDISTX 18
     6  ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   DISTX 19
     7  ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE       DISTX 20
     8  ,    XX,XDIST                                                   DISTX 21
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      DISTX 22
      EQUIVALENCE (NSEG,NAROW)                                          DISTX 23
      GO TO (100,100,5,110,110),ICONF                                   DISTX 24
    5 IF (NOSB) 100,100,120                                             DISTX 25
C            10-13-64 BYPASS OLD XX CALCULATION                         DISTX 26
  100 XX = XDIST                                                        DISTX 27
      GO TO  200                                                        DISTX 28
  110 IF (RADIUS) 100,100,120                                           DISTX 29
  120 XX = RADIUS * 2.                                                  DISTX 30
      NAMELIST/XXDIST/  XX,XDIST,RADIUS,ICONF,NOSB
  200 IF (NDBUG.GE.2) WRITE (6,XXDIST)
      RETURN
      END                                                               DISTX 32
```

```
      SUBROUTINE ECKERT                                                 ECKRT  4
CECKERT                                                                 ECKRT  1
C         FLAT PLATE,  WEDGE, CONE    CALCULATIONS                      ECKRT  2
C - - -      MARCH 27 1969      JOHN KOHLENBERGER          FORTRAN IV   ECKRT  3
C     SUBROUTINE ECKERT (ICONF,IREC,XX,REFMU,REFRHO,VSHK,RECI,WALLI,    ECKRT  5
C    1QSUBI,HSUBI)                                                      ECKRT  6
      DIMENSION   TIME1(100),ALTI(100),VINF1(100),AOFA1(100),HINSD1(100)ECKRT  7
     1  ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)    ECKRT  8
     2  ,    QI(100),QADI(100), CP(40),EK(40),FL(40),RHO(40),TEMP(40)   ECKRT  9
     3  ,    A(4,40),B(4,40)                                            ECKRT 10
     4  ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)     ECKRT 11
      COMMON      A,ALTI,AOFA1,B, CP,CPG, EK,EL, HINSD1, QADI,QCONS1,QI ECKRT 12
     1  ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                 ECKRT 13
     2  ,    AA, BB, CC,CP2, DK, T2                                     ECKRT 14
      COMMON      ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV              ECKRT 15
     1  ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                       ECKRT 16
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,              KASE,KFUNT,    NPQECKRT 17
     3  ,    NBCOL,NDBUG,NDIM,NOSB,NPGAS,NGI,NPREST,NPU,NSEG,NSEG1,NSEG2 ECKRT 18
     4  ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  ECKRT 19
     5  ,    RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLECKRT 20
     6  ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   ECKRT 21
     7  ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE       ECKRT 22
     8  ,    XX,XDIST                                                   ECKRT 23
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      ECKRT 24
      EQUIVALENCE (NSEG,NAROW)                                          ECKRT 25
      DIMENSION YCC(20),Z(20)                                           ECKRT 26
      DATA Z/.043,.045,.05,.07,.1,.2,.3,.5,1.,2.,3.,5.,6.,8.,10.,20.,
     1 100.,1000./, YCC/3.,2.8,2.7,2.,1.6,.94,.8,.73,.7,.6999,.5,.4,.32,
     2 .28,.26,.25,.22,.16,.07,.01/
    5 GO TO (10,10,10,200,200),ICONF                                    ECKRT 69
   10 GC = 32.2                                                         ECKRT 70
      GO TO (20,30),IREC                                                ECKRT 71
C             -  LAMINAR  EQUATIONS  -                                  ECKRT 72
   20 ALPHA = ANGLE/57.3                                                ECKRT 73
      IF (RADIUS.LE.0.) GO TO 1009                                      ECKRT 75
      ZE= ALPHA**2 *COS(ALPHA)/ (.802*RADIUS)* XDIST                    ECKRT 74
 1000 IF (ZE) 1009,1009,1001                                            ECKRT 76
 1001 YCV = FINTRP(Z,YCC,ZE,20,2)                                       ECKRT 77
C           NORMAL SHOCK - GRIFFITH AND LEWIS SOLUTION                  ECKRT 78
      AIINF  =.24*TINF                                                  ECKRT 79
      AISTAR = STAGI/6. *(1.+3.*WALLI/STAGI)                            ECKRT 80
      CS2    = 2.58* AIINF**.253/AISTAR**.386                           ECKRT 81
      UINF =.173E-5 + .00833E-5*AIINF                                   ECKRT 82
```

```
      REINF= RHOINF*VINF/  UINF    *RADIUS*64.4                     ECKRT 83
      IF(ZE-1.5) 1007,1007,1006                                      ECKRT 84
 1006 CSUBP = 2.46 * ALPHA**2                                        ECKRT 85
      GO TO 1008                                                     ECKRT 86
 1007 CSUBP=2.*ALPHA**2*(((((-6.3*ZE+29.3)*ZE-52.2)*ZE+43.15)*ZE-15.5)  ECKRT 87
     1 *ZE  +2.67)                                                   ECKRT 88
 1008 CH = YCV*AMINF*ALPHA**2*(1.+1.428/(AMINF**2*CSUBP))*CS2/       ECKRT 89
     1        (SQRT(REINF)*.6320)                                    ECKRT 90
      HSUBI = CH*1.4*PINF*AMINF**2*115920./VINF                      ECKRT 91
      QSUBI = HSUBI*(STAGI-WALLI)                                    ECKRT 92
      GO TO 200                                                      ECKRT 93
C             BLASIUS SOLUTION                                       ECKRT 94
 1009 REFRN = (REFRHO*VSHK*XX)/REFMU                                 ECKRT 95
      CF = .664/(SQRT (REFRN))                                       ECKRT 96
      ST = (CF/2.0)*(PR**(-2./3.))                                   ECKRT 97
      HSUBW = ST*REFRHO*VSHK*GC*3600.                                ECKRT 98
      GO TO 100                                                      ECKRT 99
C              -  TURBULENT EQUATIONS -                              ECKRT100
CHANGED  FOR  OS/360  -  NO DIVIDE BY ZERO                           ECKR 100
   30 IF (REFMU.NE.0.) GO TO 31                                      ECKR 101
      HSUBW = 0.                                                     ECKR 102
      GO TO 100                                                      ECKR 103
   31 REFRN = (REFRHO*VSHK*XX)/REFMU                                 ECKR 104
C             SCHULTZ-GRUNOW                                         ECKRT102
      CF = .370/(((ALOG(REFRN))/2.303)**2.584)                       ECKRT103
      ST = (CF/2.)*(PR**(-2./3.))                                    ECKRT104
      HSUBW = ST*VSHK*REFRHO*GC*3600.                                ECKRT105
  100 GO TO (110,110,120),ICONF                                      ECKRT106
  110 HSUBI = HSUBW                                                  ECKRT107
      GO TO 150                                                      ECKRT108
  120 GO TO (130,140),IREC                                           ECKRT109
  130 HSUBI = 1.73*HSUBW                                             ECKRT110
      GO TO 150                                                      ECKRT111
  140 HSUBI = 1.176*HSUBW                                            ECKRT112
  150 QSUBI = HSUBI*(RECI-WALLI)                                     ECKRT113
  200 RETURN                                                         ECKRT114
      END                                                            ECKRT115
```

```
      SUBROUTINE ENERGY                                                 ENCO   4
C - - -     MARCH 27 1969      JOHN KOHLENBERGER         FORTRAN IV     ENCU   3
C     SUBROUTINE ENERGY (QCONS,QSUBI,EMIS,TWALL,QNET)                   ENCO   2
CENERGY                                                                 ENCO   1
C       ENCO    DECK,LIST       ENERGY CONVERSION ROUTINE    2/21/67    ENCO   0
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)ENCO   5
     1  ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)     ENCO   6
     2  ,   QI(100),QADI(100), CP(40),FK(40),EL(40),RHO(40),TEMP(40)    ENCO   7
     3  ,   A(4,40),B(4,40)                                             ENCO   8
     4  ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)      ENCO   9
      COMMON       A,ALT1,AOFA1,B, CP,CPG, FK,EL, HINSD1, QAD1,QCONS1,QI ENCO  10
     1  ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                  ENCO  11
     2  ,   AA, BB, CC,CP2, DK, T2                                      ENCO  12
      COMMON       ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV             ENCO  13
     1  ,   EMIS,FTA1,ETA2, HINSD,HOABL,HSUBI,HV                        ENCO  14
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,           KASE.KFUNT,   NPQENCO  15
     3  ,   NBCOL,NDBUG,NDIM,NOSB,NPGAS,NCI,NPREST,NPU,NSEG,NSEG1,NSEG2 ENCO  16
     4  ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   ENCO  17
     5  ,   RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLENCO  18
     6  ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR    ENCO  19
     7  ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE        ENCO  20
     8  ,   XX,XDIST                                                    ENCO  21
     9  ,FTATP,ETALP,HVPSTM,PSTMAX                                      ENCO  22
      COMMON /PHIL/ NPQS,QSTAR(10),PSTAGN(10)
      COMMON / SWALOW/  QINT(50),HEDGE(50),PEDGE(50),UEDGE(50),HRECOV(50ENCO  23
     11)  ,   INPCW                                                     ENCO  24
C           FOR ARCHIE OSSIN - COLD WALL INPUTS FROM ENTROPY SWALLOWING ENCO  25
      EQUIVALENCE (NSEG,NAROW)                                          ENCO  26
      EQUIVALENCE (QCALT,TRANLT)                                        ENCO  27
      ENTHLP(WA) = ((.23676E-9*WA +.11571E-4)*WA +.2272)*WA +2.2864     ENCO  28
    1 IF (INPCW.LE.0) GO TO 100                                         ENCO  29
      QIOLD = QSUBI
      AIINF = ENTHLP(TINF)                                              ENCO  30
      WALLI = ENTHLP(TWALL)                                             ENCO  31
      AMINF = VINF/ (49.01* SQRT(TINF))                                 ENCO  32
      AMSA = AMINF *SIN ((AOFA+ANGLE)/57.2958)                          ENCO  33
    2 IF (SHKI.GE.1300.) GO TO 3                                        ENCO  34
      SHKMU = .42642E-7* SHKI**.493                                     ENCO  35
      RHOSHK =.576E-4  * SHKI**(-.849) *PSHK                            ENCO  36
      GO TO 4                                                           ENCO  37
    3 SHKMU = .28428E-6* SHKI**.228                                     ENCO  38
      RHOSHK= .865E-5  * SHKI**(-.584) *PSHK                            ENCO  39
    4 IF (TRANR.GT.0.) GO TO 6                                          ENCO  40
```

```
      IF (TRANLT.LT.0.)WRITE(6,5)                                     ENCO  41
    5 FORMAT ('-TRANR AND TRANLT BOTH LESS THAN ZERO, NO G.E. TRANSITION  42
     1'/' ***  LAMINAR ASSUMED  ***')
      IF (ALT.GT.TRANLT) GO TO 8                                      ENCO  44
    6 CALL DIST                                                       ENCO  45
      REYN = VSHK *RHOSHK*XDIST/ SHKMU                                ENCO  46
      IF (REYN.GT.TRANR) GO TO 9                                      ENCO  47
    8 IREC =1                                                         ENCO  48
      GO TO 7734                                                      ENCO  49
    9 IREC =2                                                         ENCO  50
 7734 IF (IREC.EQ.IRCOLD) GO TO 442                                   ENCO  51
      WRITE (6,427)  IRCOLD,IREC                                      ENCO  52
      IRCOLD = IREC                                                   ENCO  53
  427 FORMAT ( 37H- TRANSITION (COLD WALL) HAS OCCURRED / 6H FROM  I3, ENCO  54
     1     6H  TO  I3    )                                            ENCO  55
  442 PR = .76                                                        ENCO  56
      GO TO (50,60,120) ,IREC                                         ENCO  57
   50 REC = SQRT(PR)                                                  ENCO  58
      GO TO 70                                                        ENCO  59
   60 REC = PR **(.33333)                                             ENCO  60
      GO TO 70
C                    3-28-69
      ENTRY ENCO
      WALLI = ENTHLP(TWALL)
C                    3-28-69
   70 RECI =SHKI + REC* VSHK**2 / 50061.5                             ENCO  61
      REFI =SHKI + 0.5* (WALLI-SHKI) + .22*(RECI-SHKI)                ENCO  62
  120 HR = RECI                                                       ENCO  63
      HW =  WALLI                                                     ENCO  64
      HEDG  = SHKI                                                    ENCO  65
      QCOLD = QIOLD                                                   ENCO  66
      STARI = .28*HEDG +.22*HR +.5*HW                                 ENCO  67
      IF (IREC.NE.2) GO TO 30                                         ENCO  68
      IF (STARI.GE.1300.) GO TO 10                                    ENCO  69
      EKIM = .493                                                     ENCO  70
      DEN  = .849                                                     ENCO  71
      ONEK = 5.76E-5                                                  ENCO  72
      TWOK = 4.264E-4                                                 ENCO  73
      GO TO 20                                                        ENCO  74
   10 EKIM  = .288                                                    ENCO  75
      DEN   = .584                                                    ENCO  76
      ONEK  = 8.65E-6                                                 ENCO  77
      TWOK = 2.84E-7                                                  ENCO  78
   20 CONSTC = ONEK/TWOK *PSHK*VSHK*XX                                ENCO  79
      UPPER = ALOG10(CONSTC*(REFI-.5*HW)**(-EKIM-DEN))                ENCO  80
      OWERL = ALOG10(CONSTC*(REFI)       **(-EKIM-DEN))               ENCO  81
```

```
      OWL    = ALOG10(CONSTC*(.28*HEDG+.22*HR+65.)**(-FKIM-DEN))
      SECOND=  (UPPER/OWERL)**2.584                                     ENCO  82
      THIRD =  (UPPER/OWL   )**2.584
      GO TO  35                                                         ENCO  83
   30 DEN = .178                                                        ENCO  84
      SECOND = 1.                                                       ENCO  85
      THIRD = 1.
   35 CRE =  ( 1.+ 2.27*HW/HR  /(1.+1.27*HEDG/HR) )**(-DEN)  *SECOND    ENCO  86
      CR2 =  ( 1.+  295.1 /HR  /(1.+1.27*HEDG/HR) )**(-DEN)  *THIRD     ENCO
      QSUBI = QCOLD * CRE*(1.-HW/HR)                                    ENCO  87
      QSUBI = QSUBI/(CR2*(1.-130./HR))
      IF (QCONS.NE.0.)   GO TO 190
  100 IF (NPQS.LE.0)   GO TO 150
      IF (AMINF.GT.1.09)  GO TO 130
      PSTAG =(PINF + RHOINF*VINF**2 *.5 )/2116.
      GO TO 131
  130 PSTAG = PINF*(1.2*(AMINF**2))**3.5 *(6./(7.*(AMINF**2)-1.))**2.5
     1   / 2116.
C 131 STARQ = WABDOT/QNET
C     HV = FINTRP (QSTAR,PSTAGN,STARQ,NPQS,1)
  131 HV = FINTRP (PSTAGN,QSTAR,PSTAG,NPQS,1)
  150 IF (QCONS) 151,180,190
C            GETS 'QNET' AT THIS POINT
  151 CALL CONSTQ
      GO TO 200
C            USE 'QNET' BELOW IF 'QCONS'=0.0
C                                        COMMENTED CARDS DELETED
C                                        APR. 15, '71
  180 QNET = QSUBI-(.174*EMIS*((TWALL/100.)**4))                        ENCO  97
      GO TO 200                                                         ENCO  98
  190 QNET = QCONS*QSUBI - .174*EMIS*(TWALL/100.)**4                    ENCO  99
  200 IF (NDBUG.NE.14) GO TO 250                                        ENCO 100
      WRITE (6,201) TIME,ALT,VINF,IREC,   QCONS,QSUBI,EMIS,TWALL,QNET   ENCO 101
  201 FORMAT ( 14H DEBUG-ENERGY- /23H TIME, ALT, VINF, IREC /3E16.8,I6/ ENCO 102
     1   32H QCONS, QSUBI, EMIS, TWALL, QNET    /6E16.8 )               ENCO 103
      IF (INPCW.GT.0) WRITE (6,205) INPCW,HR,HW,HEDG,QCOLD,STARI,       ENCO 104
     1 EKIM,DEN,ONEK,TWOK,SECOND,   CONSTC,UPPER,OWERL , CRE            ENCO 105
  205 FORMAT (  8H INPCW = I6 /  30H HR, HW, HEDG, QCOLD, STARI   /5E16.ENCO 106
     18 /30H EKIM, DEN, ONEK, TWOK, SECOND/ 5E16.8/ 3E16.8/ 5H CRE=E16.8ENCO 107
     2  )                                                               ENCO 108
  250 RETURN                                                            ENCO 109
      END                                                               ENCO 110
```

```
      FUNCTION     FINTRP (XA,YA,X,N,M)                                  FNTRP  1
C           M = 1 LINEAR . . . . . .  M = 2 PARABOLIC . . . M = 3 STEPFNTRP  2
      DIMENSION XA(2),YA(2)                                              FNTRP  3
      I = 1                                                              FNTRP  4
      IF ( X-XA(1) )                        90,15,5                      FNTRP  5
    5 DO 10 I=2,N                                                        FNTRP  6
      IF ( XA(I)-X )                       10,15,20                      FNTRP  7
   10 CONTINUE                                                           FNTRP  8
   90 WRITE (6,96) X,XA(1),XA(I),I,N                                     FNTRP  9
   96 FORMAT ( 16H-ERROR IN FINTRP / 7X,1HX 19X,5HXA(1) 15X,5HXA(I)      FNTRP 10
     1   15X,1HI 7X,2HN / 3E20.8,2I7 )                                   FNTRP 11
   15 FINTRP = YA(I)                                                     FNTRP 12
      GO TO                                100                           FNTRP 13
   20 IF (M-2)  30,40,15
   30 IF (XA(I).EQ.XA(I-1)) GO TO 35
      FINTRP = YA(I) - (YA(I-1)-YA(I))*(XA(I)-X)/(XA(I-1)-XA(I))
      GO TO 100
   35 FINTRP = YA(I)
      GO TO                                100                           FNTRP 16
   40 IF ( N-2 )                            90,30,41                     FNTRP 17
   41 IF ( I-N )                            22,21,90                     FNTRP 18
   21 I=I-1                                                              FNTRP 19
   22 S2 = XA(I-1)**2                                                    FNTRP 20
      S1 = S2-XA(I)**2                                                   FNTRP 21
      S2 = S2-XA(I+1)**2                                                 FNTRP 22
      X1 = XA(I-1)-XA(I)                                                 FNTRP 23
      X2 = XA(I-1)-XA(I+1)                                               FNTRP 24
      Y1 = YA(I-1)-YA(I)                                                 FNTRP 25
      Y2 = YA(I-1)-YA(I+1)                                               FNTRP 26
      FINTRP = ((Y1/X1-Y2/X2)/(S1/X1-S2/X2)*X+(Y1/S1-Y2/S2)/             FNTRP 27
     1  (X1/S1-X2/S2))*X+YA(I)-((Y1/X1-Y2/X2)/(S1/X1-S2/X2)*             FNTRP 28
     2  XA(I)+(Y1/S1-Y2/S2)/(X1/S1-X2/S2))*XA(I)                         FNTRP 29
  100 RETURN                                                             FNTRP 30
      END
```

```
                                                                        CPKK   2
      SUBROUTINE GECPK2                                                 CPKK   1
CGECPK2
C     THIS ROUTINE GETS CP2(I,K), DK(I,K) I=1,NAROW K=1,NBCOL FOR TWOD  CPKK   3
C - - -      APRIL 8 1964      JOHN KOHLENBERGER         FORTRAN IV     CPKK   4
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),ADFA1(100),HINSD1(100)CPKK   5
     1   ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)    CPKK   6
     2   ,   QI(100),QAD1(100), CP(40),FK(40),FL(40),RHO(40),TEMP(40)   CPKK   7
     3   ,   A(4,40),B(4,40)                                            CPKK   8
     4   ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)     CPKK   9
      COMMON          A,ALT1,ADFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI  CPKK  10
     1   ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                 CPKK  11
     2   ,   AA, BB, CC,CP2, DK, T2                                     CPKK  12
      COMMON          ALT,AMINF,ADFA,ANGLE, CONPR, DELTA,DQSAV          CPKK  13
     1   ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                       CPKK  14
     2   ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,            KASE,KFUNT,     NPQCPKK  15
     3   ,   NBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 CPKK  16
     4   ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  CPKK  17
     5   ,   RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLCPKK  18
     6   ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   CPKK  19
     7   ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE       CPKK  20
     8   ,   XX,XDIST                                                   CPKK  21
     9   ,ETATP,ETALP,HVPSTM,PSTMAX
      EQUIVALENCE  (NSEG,NAROW)                                         CPKK  22
C      KFUNT = 0 ,DK AND CP2 ENTERED DIRECTLY,  DEBUG RETAINED
C     KFUNT = 1 EK(I) AND CP(I) ARE BOTH FCNS OF TEMP(I)                CPKK  23
C     KFUNT = 2 FK(I) IS FCN OF TEMP(I), CP(I) ARE CONSTANT             CPKK  24
C     KFUNT = 3 FK(I) ARE CONSTANT, CP(I) ARE FCNS OF TEMP(I)           CPKK  25
C     KFUNT = 4 BOTH EK(I) AND CP(I) ARE CONSTANT                       CPKK  26
      COMMON /MIKE/ STUFF(71),RHOTBL(10,40),CPTBL(10,40),EKTBL(10,40),
     1     TEMTBL(10,40)                                                CPKK  28
      DO 510 I = 1,NAROW
      IF (KFUNT .EQ. 0)   GO TO 504
      IF (KFUNT.GT.4)  GO TO 501                                        CPKK  27
   10 GO TO (100,200,300,400),KFUNT                                     CPKK  29
  100 DO 125 K = 1,NBCOL                                                CPKK  30
  105 IF (CC(I,K))110,125,110                                           CPKK  31
  110 DK(I,K) = A(1,I)+T2(I,K)*(A(2,I)+T2(I,K)*(A(3,I)+A(4,I)*T2(I,   CPKK  32
     1K)))                                                              CPKK  33
  120 CP2(I,K) = B(1,I)+T2(I,K)*(B(2,I)+T2(I,K)*(B(3,I)+B(4,I)*       CPKK  34
     1T2(I,K)))                                                         CPKK  35
  125 CONTINUE                                                          CPKK  36
  130 GO TO 504                                                         CPKK  38
  200 DO 225 K = 1,NBCOL
```

```
  205 IF (CC(I,K))210,225,210                                          CPKK  39
  210 CP2(I,K) = B(1,I)                                                CPKK  40
  220 DK(I,K) = A(1,I)+T2(I,K)*(A(2,I)+T2(I,K)*(A(3,I)+A(4,I)*         CPKK  41
     1T2(I,K)))                                                        CPKK  42
  225 CONTINUE                                                         CPKK  43
  230 GO TO 504                                                        CPKK  44
  300 DO 325 K = 1,NBCOL                                               CPKK  46
  305 IF (CC(I,K))310,325,310                                          CPKK  47
  310 DK(I,K) = A(1,I)                                                 CPKK  48
  320 CP2(I,K) = B(1,I)+T2(I,K)*(B(2,I)+T2(I,K)*(B(3,I)+B(4,I)         CPKK  49
     1*T2(I,K)))                                                       CPKK  50
  325 CONTINUE                                                         CPKK  51
  330 GO TO 504                                                        CPKK  52
  400 DO 425 K = 1,NBCOL                                               CPKK  54
  405 IF (CC(I,K))410,425,410                                          CPKK  55
  410 DK(I,K) = A(1,I)                                                 CPKK  56
  420 CP2(I,K) = B(1,I)                                                CPKK  57
  425 CONTINUE                                                         CPKK  58
      GO TO 504
C            THIS IS NOY IN FINAL FORM, RHO2(I,KE SHOULD BE FORMED
  501 K=NSEG1
C            A ROUGH GESTIMATION FOR T2
      RHO(I) = FINTRP (TEMTBL(1,I),RHCTBL(1,I),T2(I,K),KFUNT,1)
      DO 502 K=1,NBCOL
      IF (CC(I,K).EQ.0.) GO TO 504
      CP2(I,K) = FINTRP(TEMTBL(1,I),CPTBL(1,I),T2(I,K),KFUNT,1)
      DK (I,K) = FINTRP(TEMTBL(1,I),EKTBL(1,I),T2(I,K),KFUNT,1)
  502 CONTINUE
  504 IF (NOBUG.EQ.2)  WRITE (6,505) I,RHO(I),(K,CP2(I,K),DK(I,K)
     1 , K=1,NBCOL)
  505 FORMAT ( I6, E16.8/(I6,6E16.8))
  510 CONTINUE
  500 RETURN                                                           CPKK  59
      END                                                              CPKK  60
```

```
      SUBROUTINE GETCPK                                                 CPAND   3
CGETCPK                                                                 CPAND   1
C - - -     MARCH 27 1964       JOHN KOHLENBERGER        FORTRAN IV     CPAND   2
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)CPAND   4
     1   ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)   CPAND   5
     2   ,    QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)  CPAND   6
     3   ,    A(4,40),B(4,40)                                           CPAND   7
     4   ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)    CPAND   8
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI CPAND   9
     1   ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                CPAND  10
     2   ,    AA, BB, CC,CP2, DK, T2                                    CPAND  11
      COMMON       ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV             CPAND  12
     1   ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                      CPAND  13
     2   ,ICHK,ICK,ICONF,ISO,INR4,IN342,IREC,        KASE,KFUNT,     NPQCPAND  14
     3   ,    NBCOL,NOBUG,NDIM,NOSB,NPGAS,NCI,NPPEST,NPU,NSEG,NSEG1,NSEG2 CPAND  15
     4   ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  CPAND  16
     5   ,    RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLCPAND  17
     6   ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   CPAND  18
     7   ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE       CPAND  19
     8   ,    XX,XDIST                                                  CPAND  20
     9   ,ETATP,ETALP,HVPSTM,PSTMAX                                     CPAND  21
      COMMON /MIKE/ STUFF(64),JUNK(2),SOME(4),JAZZ(2),                  CPAND  22
     3        RHOTBL(10,40),CPTBL(10,40),EKTBL(10,40),TEMTBL(10,40)
      EQUIVALENCE (NSEG,NAROW)                                          CPAND  24
C     KFUNT = 1 EK(I) AND CP(I) ARE BOTH FCNS OF TEMP(I)                CPAND  25
C     KFUNT = 2 EK(I) IS FCN OF TEMP(I), CP(I) ARE CONSTANT             CPAND  26
C     KFUNT = 3 EK(I) ARE CONSTANT, CP(I) ARE FCNS OF TEMP(I)           CPAND  27
C     KFUNT = 4 BOTH EK(I) AND CP(I) ARE CONSTANT                       CPAND  28
      IF (KFUNT.GT.4) GO TO 5678                                        CPAND  29
C          TABLE LOOK-UP  - TEMP DEPENDANT RHO,CP,EK        AT 5678     CPAND  30
   10 GO TO (100,200,300,400),KFUNT                                     CPAND  31
C     NOTE - POSSIBLY BYPASS COMPUTING OF CONSTANTS AFTER 1ST STEP      CPAND  32
  100 DO 120 I = NSEG2,NSEG                                             CPAND  33
      EK(I) = A(1,I)+TEMP(I)*(A(2,I)+TEMP(I)*(A(3,I)+A(4,I)*TEMP(I)))   CPAND  34
  120 CP(I) = B(1,I)+TEMP(I)*(B(2,I)+TEMP(I)*(B(3,I)+B(4,I)*TEMP(I)))   CPAND  35
      GO TO 500                                                         CPAND  36
  200 DO 220 I = NSEG2,NSEG                                             CPAND  37
      CP(I) = B(1,I)                                                    CPAND  38
  220 EK(I) = A(1,I)+TEMP(I)*(A(2,I)+TEMP(I)*(A(3,I)+A(4,I)*TEMP(I)))   CPAND  39
      GO TO 500                                                         CPAND  40
  300 DO 320 I = NSEG2,NSEG                                             CPAND  41
      EK(I) = A(1,I)                                                    CPAND  42
  320 CP(I) = B(1,I)+TEMP(I)*(B(2,I)+TEMP(I)*(B(3,I)+B(4,I)*TEMP(I)))   CPAND  43
```

```
      GO TO 500                                                         CPAND 44
      DO 420 I = NSEG2,NSEG                                             CPAND 45
      EK(I) = A(1,I)                                                    CPAND 46
  420 CP(I) = B(1,I)                                                    CPAND 47
  500 RETURN                                                            CPAND 48
 5678 DO 5680 I=NSEG2,NSEG                                              CPAND 49
      RHO(I) = FINTRP(TEMTBL(1,I),RHOTBL(1,I),TEMP(I),KFUNT,1)          CPAND 50
      EK(I)  = FINTRP(TEMTBL(1,I), EKTBL(1,I),TEMP(I),KFUNT,1)          CPAND 51
      CP(I)  = FINTRP(TEMTBL(1,I), CPTBL(1,I),TEMP(I),KFUNT,1)          CPAND 52
      IF (NDBUG.EQ.1) WRITE(6,5679) I,RHO(I),EK(I),CP(I)                CPAND 53
 5679 FORMAT ( 28H RHO,K,CP INTERPOLATED,SEG=  I3,3E16.8 )              CPAND 54
 5680 CONTINUE                                                          CPAND 55
      GO TO 500                                                         CPAND 56
      END                                                               CPAND 57
```

```
      SUBROUTINE HANKEY                                                 FLUX   6
CHANKEY  -  SONIC POINT HEAT FLUX SUBROUTINE     -AEROSPACE CORP.(SBO)  FLUX   1
C                 COMPOSED BY LEO BERGER                                FLUX   2
C                 WRITTEN  BY JOHN KOHLENBERGER                         FLUX   3
C     3-24-64     TO THE TUNE OF SMITH-FISHER  (3006A)                  FLUX   4
C      AS DIRECTED BY Z. SVETVAY                                        FLUX   5
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)FLUX   7
     1 ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)      FLUX   8
     2 ,   QI(100),QADI(100), CP(40),FK(40),EL(40),RHO(40),TEMP(40)     FLUX   9
     3 ,   A(4,40),B(4,40)                                              FLUX  10
     4 ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)       FLUX  11
      COMMON      A,ALT1,AOFA1,B, CP,CPG, FK,EL, HINSD1, QADI,QCONS1,QI FLUX  12
     1 ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                   FLUX  13
     2 ,   AA, BB, CC,CP2, DK, T2                                       FLUX  14
      COMMON      ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV              FLUX  15
     1 ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                         FLUX  16
     2 ,ICHK,ICK,ICONE,IGO,INB4,INB42,IRTC,          KASE,KFUNT,    NPQFLUX  17
     3 ,   NACUL,NOPUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 FLUX  18
     4 ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   FLUX  19
     5 ,   RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REY I,RHOINF,RHOSHK,RHOSLFLUX  20
     6 ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR     FLUX  21
     7 ,   TSHK,TWALL,TWNP, VINF,VSHK, W,WAB,WABDOT,WALLI,WR TE         FLUX  22
     8 ,   XX,XDIST                                                     FLUX  23
     9 ,ETATP,ETALP,HVPSTM,PSTMAX                                       FLUX  24
      EQUIVALENCE (NSEG,NAROW),(QCALT,TRANLT)                           FLUX  25
      RN = XX/2.                                                        FLUX  26
      IF (RADIUS.GT.0.) GO TO 399                                       FLUX  27
      RADIUS = RN                                                       FLUX  28
  399 IF (ICONE.LT.3)    GO TO 400                                      FLUX  29
C           BELOW ,      SHKMU IS REALLY MU-INFINITY                    FLUX  30
      XX = RADIUS *2.                                                   FLUX  31
  400 AIINF =    ((.23676E-9*TINF+.11571E-4)*TINF+.2272)*TINF+2.2864    FLUX  32
      SHKMU = .42642E-7 *(AIINF**.493)                                  FLUX  33
      REYN  = VINF*RHOINF*RADIUS/SHKMU                                  FLUX  34
      VSHK = .301*VINF                                                  FLUX  35
      IF (TRANR) 426,426,429                                            FLUX  36
  426 IF (ALT-TRANLT) 427,428,430                                       FLUX  37
  427 IF (TRANR) 428,428,429                                            FLUX  38
  428 CALL PRINT                                                        FLUX  39
      WRITE (6,666)                                                     FLUX  40
      TRANR = REYN                                                      FLUX  41
      GO TO  435                                                        FLUX  42
  429 IF (REYN-TRANR)430,430,435                                        FLUX  43
```

```
C          CHANGED 8-4-67 TO GUARD AGAINST OS/360 DIVIDE BY O.              FLUXG 43
  430 IF (RADIUS.LE.0.) GO TO 436                                          FLUXG 44
      QDOT = 10800.*SQRT(RHOINF/RHOSL)*((VINF/26000.)**3.15)               FLUXG 45
     1      *((STAGI-WALLI)/(STAGI-129.5))/SQRT(RADIUS)                    FLUXG 46
      GO TO 440                                                            FLUXG 47
  435 IF ((STAGI.EQ.0.).OR.(RN.LE.0.)) GO TO 436                           FLUXG 48
      QDOT = 5.*(RHOINF/RHOSL)**.8*(VINF/1000.)**3*(1.-WALLI/STAGI)        FLUXG 49
     1      /(RN**.2)                                                      FLUXG 50
      GO TO 440                                                            FLUXG 51
  436 QDOT = 0.                                                            FLUXG 52
  440 QSUBI = QDOT*3600.                                                   FLUX  49
  666 FORMAT ( 53H-* * * * * * HANKEY TRANSITION HAS OCCURRED * * * * *)FLUX  50
      GO TO 600                                                            FLUX  51
C 500 CALL KEMP                                                            FLUX  52
  600 IF (NOBUG-8) 700,610,700                                             FLUX  53
  610 WRITE (6,612) QDOT,QSUBI,RHOINF,VINF,RN,RHOSL                        FLUX  54
  612 FORMAT (61H-.HANKEY. DEBUG,   INCLUDES  QDOT,QSUBI,RHOINF,VINF,RN,FLUX  55
     1 RHOSL      / 6F16.8 )                                               FLUX  56
  700 RETURN                                                               FLUX  57
      END                                                                  FLUX  58
```

```
      SUBROUTINE INPUT                                                  INPU   0
C - - -    OCTOBER 14 1970    JOHN KOHLENBERGER        FORTRAN IV       INPU   1
CINPUT B006F, USING I.B.M. NAMELIST    FORTRAN IV (7094 , 360 EMULATOR) INPU   2
C            MODIFIED FOR FORTRAN IV-G, VERSION 14 COMPILER             INPU   3
C - - -   JANUARY 20,1968   JOHN KOHLENBERGER   AEROSPACE CORP. SAN BERINPU   4
C          UP-DATED JUNE 11,1968                                        INPU   5
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)INPU   6
     1  ,   TINSD1(100),QCONS1(100),VAPPES(100),CPG(100),TEMPW(100)     INPU   7
     2  ,   QI(100),QA01(100), CP(40),EK(40),FL(40),RHO(40),TEMP(40)    INPU   8
     3  ,   A(4,40),B(4,40)                                             INPU   9
     4  ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)      INPU  10
      DIMENSION  TIME(100),ALT(100),VINF(100),AOFA(100),                INPU  11
     1               QCONS(100),STAGEN(100),TEMPER(40)                  INPU  12
      DIMENSION A1(40),A2(40),A3(40),A4(40),B1(40),B2(40),B3(40),B4(40) INPU  13
      DIMENSION   GARBIJ(20)                                            INPU  14
      DIMENSION DATA(20)                                                INPU  15
C                                                                       INPU  16
      COMMON      A,ALT1,AOFA1,B, CP,CPG, EK,FL, HINSD1, QA01,QCONS1,QI INPU  17
     1  ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPPES,VINF1                  INPU  18
     2  ,   AA, BB, CC,CP2, DK, T2                                      INPU  19
      COMMON ALTUD,AMINF,ANGOFA,ANGLE,     CONPR,       DELTA,DOSAV     INPU  20
     1  ,EMIS,ETA1,ETA2,        HINSD ,HOABL,HSUBI,HV                   INPU  21
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,            KASE,KFUNT,    NPQINPU  22
     3  ,MBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPJ,NSFG,NSFG1,NSEG2    INPU  23
     4  ,NTRAJ,    PINF,PR,PSHK,PSL, QCALT,QCONST,QINSD,QNET,QSAV,QSUBI INPU  24
     5  ,RADIUS,EFCI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSL,  INPU  25
     6  TSHKI,SHKMJ,     STOP,SWEEP,STAGI, TABL,TIM,TINF,TINSD ,TRANR   INPU  26
     7  ,TSHK,TWALL,TWNB,      VELINF,VSHK,       W,WAB,WABDOT,WALLI,WRITEINPU  27
     8  ,   XX,XDIST                                                    INPU  28
     9     ,ETATP,ETALP,HVPSTM,PSTMX,NT
      COMMON /BLOK1/ TITLE(20)                                          INPU  30
      COMMON / SWALOW/  QINT(50),HEDGE(50),PEDGE(50),UEDGE(50),HRECOV(50INPU  31
     1)  ,  INPCW                                                       INPU  32
C            FOR ARCHIE USSIN - COLD WALL INPUTS FROM ENTROPY SWALLOWING INPU  33
      COMMON / MIKE / HV2,ETAL2,ETAT2,XI, FV1P(10),FV1S(10),FV2P(10),   INPU  34
     *      FV2S(10),STGTBL(10),SHRTBL(10),NOPSTG,NOPSHR                INPU  35
     2   ,  QSONIC,SHEAR,HOABLM,HE, ISTETS, NOCHEK,                     INPU  36
     3      RHOTBL(10,40),CPTBL(10,40),EKTBL(10,40),TEMTBL(10,40)       INPU  37
      COMMON /WERNER/   TRVRSL,TFB(100),RHOFB(100),PFB(100),ALTCM(100)  INPU  38
      COMMON /CHARAB/ NTME,TSTOR(50),QO(50),PO(50),HSTAB(50),HLTAB(50), INPU  39
     1 UETAB(50)                                                        INPU  40
      COMMON /PHIL/ NPQS,QSTAR(10),PSTAGN(10)
      COMMON /RALPH/ TWTAB(20),EMTAB(20),NEMIS,NQCOR,QABSF(20),TWA(20)
```


Reproduced from best available copy.

```
C                                                                       INPU  41
      EQUIVALENCE (NSEG,NAROW),(QCALT,TRANLT)                           INPU  42
C                                                                       INPU  43
      NAMELIST/BD060S/  KASE,ICONF,NDIM,NTRAJ,KFUNT,NSEG,NBCOL,NPU,ICHK,INPU  44
     1  CONPR,TRANR,TRANLT,ANGLF,SWEEP,XDIST,EMIS,DELTA,WRITE,STOP,STARTINPU  45
     2 ,TABL,HOABL,HV,ETAL,ETAT,NOSB,RADIUS,TWALL,RHO,A,B,TIME,ALT,VINF,INPU  46
     3  AOFA,HINSD1,TINSD1,QCONS,NPQ,STAGEN,TEMPW,TEMP,EL,QI,QSOLAR,    INPU  47
     4 NDBUG,ISTFTS,TEMTBL,CPTBL,EKTBL,RHCTBL                           INPU  48
     4  ,A1,B1,A2,B2,A3,B3,A4,B4,NQI,IHANK,NPGAS,PSTMAX,HVPSTM,ETATP    INPU  49
     5  ,ETALP,    INPCW,QINT,HEDGE,PEDGE,UEDGE,HRECOV                  INPU  50
     6   ,    HV2,ETAL2,ETAT2,XI,FV1P,FV1S,FV2P,FV2S,STGTBL,SHRTBL,NOPSTG INPU  51
     7  ,     NOPSHR,HF,TITLE,TRVRSL,TFB,RHCFB,PFB,ALTCM , NOCHEK       INPU  52
     8 ,NPQS,QSTAR,PSTAGN,TWTAB,EMTAB,NEMIS,NQCOR,QABSF,TWA
      NAMELIST /TWODIM/ AA,BB,CC,T2, NAROW,NBCOL ,DK,CP2                INPU  53
C                                                                       INPU  54
      DATA IHANK/0/                                                     INPU  55
      DATA A1/40*0./,A2/40*0./,A3/40*0./,A4/40*0./                      INPU  56
     1     ,B1/40*0./,B2/40*0./,B3/40*0./,B4/40*0./                     INPU  57
      DATA  CKEND,WRONG,END /'    $','060S',' END'/                     INPU  58
      DATA TERM/' &EN'/,JAY/0/                                          INPU  59
      DATA TEMPER/40*540./,START/0./                                    INPU  60
C                                                                       INPU  60
10000 IF (IGO.GT.0) GO TO 28                                            INPU  61
C            IGO = - , FIRST TIME THROUGH                               INPU  61
C            IGO = -0, TO READ A NEW CASE                               INPU  61
C            IGO = + , TO RESTART SAME CASE                             INPU  61
      CALL AEROSP                                                       INPU  62
      WRITE (6,1967)                                                    INPU  63
 1967 FORMAT(19X,' &&&&  BBBB   DDDD    000     666   000    SSSS'/     INPU  64
     1          19X,'&      B   B  D   D  0   0  6      0 0 0  S     '/ INPU  65
     2          19X,' &&&&  BBBB   D   D  0   0  6 66   0 0 0   SSS '/  INPU  66
     3          19X,'&      B   B  D   D  0   0  6   6  0 0 0      S'/  INPU  67
     4          19X,' &&&&  BBBB   DDDD    000     666    000    SSSS '// INPU  68
     5 19X,' AEROSPACE SYNTHESISIZED AERODYNAMIC HEATING - ASAH',5X,    INPU  69
     6 ' FOR IBM 360-65(FORTRAN H-2) 4/15/71')
      NDBUG1 = NDBUG                                                    INPU  71
      DELTA = DELTSV                                                    INPU  72
      NSEG1 = 1                                                         INPU  73
      NSEG2 = 1                                                         INPU  74
C            WRITE OUT CARD IMAGES   5/19/66  J.M.K                     INPU  75
      IF (JAY.GE.1) GO TO  8                                            INPU  76
C           JAY = NO. CARDS FROM 5 TO 9                                 INPU  76
 1776 REWIND 9                                                          INPU  77
      NCARD=0                                                           INPU  78
      ICASE=0                                                           INPU  79
   66 FORMAT (1H1,' COMPLETE LIST OF ALL INPUT CARDS TO BE PROCESSED '/)INPU  80
```

```
      WRITE (6,66)                                                      INPU  91
 4444 JAY = 1                                                           INPU  92
 4445 READ (5,5555) DATA                                                INPU  83
 5555 FORMAT (20A4)                                                     INPU  84
      WRITE (6,5555)  DATA                                              INPU  85
      IF (DATA(1).NE.TERM) GO TO 7730                                   INPU  86
      ICASE = ICASE + 1                                                 INPU  87
      WRITE (6,7731) ICASE                                              INPU  88
 7731 FORMAT ( 85X,  'END CASE',I6)                                     INPU  89
 7730 IF  ((DATA(20).EQ.CKEND) .OR. (DATA(1).EQ.END))GO TO 888          INPU  90
C           SINGLE 'ENDDATA' WILL TERMINATE DATA CASE GROUPE            INPU  91
C            '   $' AT END NOW TERMINATES ALL READING, OR 'END 'AT FIRST INPU  92
      JAY = JAY + 1                                                     INPU  93
      IF (DATA(2) .EQ. WRONG) GO TO 7734                                INPU  94
      WRITE (9,5555) DATA                                               INPU  95
      GO TO 4445                                                        INPU  96
 7734 WRITE (6,7735)                                                    INPU  97
 7735 FORMAT ('BAD NAMELIST NAME, SHOULD BE BD060S, NOT B0060S ' )      INPU  98
      GO TO 4445                                                        INPU  99
  888 WRITE (6,9) NOBUG                                                 INPU 100
      ICASE = 0                                                         INPU 101
      JAY = JAY-1                                                       INPU 102
      END FILE 9                                                        INPU 103
      WRITE (6,889) JAY                                                 INPU 104
  889 FORMAT ( I6,' CARDS READ')                                        INPU 105
      IF (JAY.LE.0) STOP                                                INPU 106
C           DOUBLE 'ENDDATA' WILL TERMINATE JOB                         INPU 107
      NCARD = JAY                                                       INPU 108
      REWIND 9                                                          INPU 109
    3 IF (NCARD.LT. 1 ) GO TO 883                                       INPU 110
C           NCARD = NUMBER OF CARDS LEFT ON 9                           INPU 110
      ICASE = ICASE + 1                                                 INPU 111
      WRITE (6,6) ICASE                                                 INPU 112
    6 FORMAT (1H1,' INPUT CARDS READ, DATA SET' I6)                     INPU 113
      KARD=0                                                            INPU 114
  887 READ (9,5555) DATA                                                INPU 115
      KARD =KARD+1                                                      INPU 116
      WRITE (6,5555) DATA                                               INPU 117
      IF (DATA(1).EQ.TERM) GO TO 886                                    INPU 118
      GO TO 887                                                         INPU 119
  883 WRITE (6,884)                                                     INPU 120
  884 FORMAT ('1ALL INPUT CARDS HAVE BEEN PROCESSED ')                  INPU 121
      CALL AEPOSP                                                       INPU 122
      GO TO 1776                                                        INPU 123
  886 DO 885 I=1,KARD                                                   INPU 124
      BACKSPACE 9                                                       INPU 125
```

```
  885 CONTINUE                                                             INPU 126
      NCARD = NCARD - KARD                                                 INPU 127
      WRITE (6,881) KARD,NCARD                                             INPU 128
  881 FORMAT (' KARD = ',I6,',    TO GC =' I6)                             INPU 129
C             END OF CARD IMAGES       5/19/66  J.M.K.                     INPU 130
C                                   FOR ONE DIM. ONLY                      INPU 131
   28 INB4 = 0                                                             INPU 132
C     1 DIM INPUT                                                          INPU 133
C     ICONF = 1 FLAT PLATE                                                 INPU 134
C             = 2 WEDGE                                                    INPU 135
C             = 3 CONE                                                     INPU 136
C             = 4 SPHERE                                                   INPU 137
C             = 5 CYLINDER                                                 INPU 138
C     NTRAJ = NO. OF TRAJECTORY CARDS (MAXIMUM OF 100 )                    INPU 139
C     KFUNT = 1 EK(I) AND CP(I) ARE BOTH FCNS OF TEMPERATURE               INPU 140
C             = 2 EK(I) IS A FCN OF TEMPERATURE, CP(I) IS CONSTANT         INPU 141
C             = 3 EK(I) IS CONSTANT, CP(I) IS A FCN OF TEMPERATURE         INPU 142
C             = 4 BOTH ARE CONSTANT                                        INPU 143
C     NSEG  =NAROW = NUMBER OF SEGMENTS(MAXIMUM OF 20 IN 1D, 10 IN 2D )   INPU 144
C     NO. SEGMENTS - OR SLABS  USED IN THIS PROBLEM                        INPU 145
C     NPOINT= NO. POINTS IN THOSE INDIVIDUAL SLABS                         INPU 146
C           NDBUG IS FOR DEBUGGING PURPOSES ONLY                           INPU 147
C     NDBUG = C  NO DEBUG PRINT                                            INPU 148
C          NDBUG VALUES FOR ROUTINES, ARE LISTED BELOW
C           .EQ.1    ABLATE
C           .EQ.1    GETCPK
C           .GE.1    MAIN
C           .GE.1    ABLUNT
C           .GE.1    KEMP
C           .EQ.2.    DIST
C           .EQ.2    GECPK2
C           .EQ.3 ATMOS
C           .EQ.3    TRAJ
C           .EQ.3    PME
C           .EQ.5    THERMO
C           .EQ.4    SHOCK
C           .EQ.8    HANKEY
C           .EQ.10   ONED
C           .EQ.14   ENERGY/CONSTO
C           .GE.14   TWOD
C 11/22/70 THESE MAY BE ADDED TO, OR CHANGED, WITHOUT NOTICE AT ANY TIME
      IF (IGO)  1,2,3                                                      INPU 150
C     NPU   = 1 FOR PUNCH OPTION - USED FOR CHARRING ABLATOR INPUTS        INPU 151
C     NQI   = QI CALC. IN -ABLATE- WHERE, DOSAV=F(QUSE,EL,EK,TEMP,DELTA)INPU 152
C             = - OR 0 ,  QUSE = QNET                                      INPU 153
C             = + , QUSE = QSUBI*QCONS                                     INPU 154
```

```
      C     ICHK  = 0 , FOR REGULAR PROGRAM FLOW                                  INPU 155
      C           = + FOR HANKEY CALC.                                            INPU 156
      C     CONPR = CONE PRESSURE  AS USED IN *PME*           10/23/63            INPU 157
      C     TRANR = TRANSITION REYNOLDS NO.                                       INPU 158
      C           TRANLT  IS TURB.-LAM.  ALTITUDE                                 INPU 159
      C     ANGLE = WEDGE OR CONE  HALF ANGLE                                     INPU 160
      C     SWEEP = SWEEP ANGLE                                                   INPU 161
      C     XDIST = DISTANCE FROM LEADING EDGE FOR FLAT PLATE, WEDGE OR CONE      INPU 162
      C           = DIAMETER FOR SPHERE OR CYLINDER                       (FEET)  INPU 163
      C     EMIS  = SURFACE EMISSIVITY                                            INPU 164
      C     DELTA = TIME STEP SIZE         ( SECONDS )                            INPU 165
      C     WRITE = PRINT OUT TIME INTERVAL      ( SECONDS )                      INPU 166
      C     STOP  = END OF FLITE TIME                                             INPU 167
      C          START = ALT = START TIME , IF OTHER THAN ZERO                    INPU 168
      C     TABL = ABLATION TEMPERATURE ( DEGREES R )                             INPU 169
      C     HOABL = HEAT OF ABLATION   ( BTU/LB   )                               INPU 170
      C     OLD  HOABL NOW =  HV + ETA(1OR2)* (RECI - WALLI)                      INPU 171
      C     HOABL CAN = HV ,IF A CONSTANT VALUE IS DESIRED                        INPU 172
      C     NOSB    =  IF POSITIVE, USE NOSE BLUNTING ROUTINE (ABLUNT)            INPU 173
      C     RADIUS =  INITIAL RADIUS OF BLUNT NOSE                                INPU 174
      C     TWALL  =  CONSTANT WALL TEMP USED IN NOSE BLUNTING ROUTINE (DEG R)INPU 175
      C     RHO(I) = SPECIFIC WT. OF SEGMENT I       ( LBS/FT**3  )               INPU 176
      C     A(I,J) = COEFFICIENTS OF JTH SEGMENT FOR COMPUTING EK(J) THUSLY       INPU 177
      C              EK(J) = A(1,J) + A(2,J)*T + A(3,J)*T**2 + A(4,J)*T**3        INPU 178
      C     B(I,J) = ARE TO CP(J)  AS THE A S ARE TO EK(J)                        INPU 179
          1 NDBUG1 = 0                                                            INPU 180
      C     TRAJECTORY INPUT                                                      INPU 181
      C     TIME =  TIME ( SECONDS )                                              INPU 182
      C     ALT  =  ALTITUDE ( FEET )                                             INPU 183
      C     VINF =  VELOCITY (FT/SEC )                                            INPU 184
      C     AOFA =  ANGLE OF ATTACK ( DEGREES )                                   INPU 185
      C     HINSD = INSIDE HEAT TRANSFER  ( BTU/HR FT**2   DEGREE R  )            INPU 186
      C     TINSD = INSIDE TEMPERATURE ( DEGREES R )                              INPU 187
      C     IF THE QCONS(I) ARE NOT READ IN THEY SHOULD ALL = 1                   INPU 188
      C           IF NPQ =-NO.,HV IS FOUND FROM STAGEN VS. TEMPW CURVE,           INPU 189
      C            WHERE, STAGEN = STAGNATION ENTHALPY                            INPU 190
      C            AND    TEMPW  = HEAT OF VAPORIZATION (HV)                      INPU 191
      C     TEMP(I) = TEMPERATURE OF ITH SEGMENT          ( DEGREES R )           INPU 192
      C     EL(I) = THICKNESS OF ITH SEGMENT     ( INCHES )                       INPU 193
      C           NOTE . . .  QI STORED IN QADI ,  .OUTPUT. STORES QI*3600=QI     INPU 194
      C     NOCHEK = 0, TO CHECK NEG. TEMPS. AND STABILITY                        INPU 195
      C            = 1, FOR NEGATIVE TEMPERATURE CHECK ONLY                       INPU 196
      C            = 2, FOR NO CHECKS                                             INPU 197
      C                                                                           INPU 198
      C     2 DIM INPUT                                                           INPU 199
      C     NAROW = NUMBER OF ROWS IN 2-D MESH (MAX.=10)                          INPU 200
```

```
C     NBCOL = IN 2 DIMENSIONS IS NO. CFCOLUMS (MMAX.=10)                 INPU 201
C     AA(J) = THICKNESS OF J-TH SEGMENT (INCHES)                         INPU 202
C     BB(I) = WIDTH OF ITH SEGMENT IN 2D.                       (INCHES) INPU 203
C         T2(J,I) = TEMPERATURE (ROW,COLUMN)                             INPU 204
C     CP2(J,I) = SPECIFIC HEAT    (ONLY IF KFUNT=0)                      INPU 205
C      DK(J,I) = CONDUCTIVITY     (ONLY IF KFUNT=0)                      INPU 206
C                                                                        INPU 207
      IHANK = 0                                                          INPU 208
      START = 0.0                                                        INPU 209
      NPQ = 0                                                            INPU 210
      NOSR= 0                                                            INPU 211
C     RADIUS=0.                                                          INPU 212
      RAD1 = RADIUS                                                      INPU 213
      TWALL=0.                                                           INPU 214
      TRANR=0.                                                           INPU 215
      NPGAS=0                                                            INPU 216
      NPQS = 0
      DO 396 I=1,10
      QSTAR(I) = 0.0
  396 PSTAGN(I) = 0.0
      DO 96 I=1,100                                                      INPU 217
      IF (NPQ.LE.1) GO TO 96                                             INPU 218
      QI(I) = QAD1(I)                                                    INPU 219
      IF (I.GT.40) GO TO 96                                              INPU 220
      TEMPER (I) = TEMP(I)                                               INPU 221
      TIME(I) = TIME1(I)                                                 INPU 222
      ALT(I) = ALT1(I)                                                   INPU 223
      VINF(I) = VINF1(I)                                                 INPU 224
   96 QCONS(I) = 1.0                                                     INPU 225
    2 DO 99 I=1,100                                                      INPU 226
      STAGEN(I) = VAPRES (I)                                             INPU 227
      AOFA(I) = AOFA1(I)                                                 INPU 228
      ALT(I) = ALT1(I)                                                   INPU 229
      TIME(I) = TIME1(I)                                                 INPU 230
      QCONS(I) = QCONS1(I)                                               INPU 231
      QI(I) = QAD1(I)                                                    INPU 232
      IF (I.GT.40) GO TO 99                                              INPU 233
      TEMP(I) = TEMPER(I)                                                INPU 234
   99 VINF(I) = VINF1(I)                                                 INPU 235
      IF (NPQ.LE.1) GO TO 98                                             INPU 236
      DO 97  JETHRO = 1,NPQ                                              INPU 237
   97 QI(JETHRO) = QAD1(JETHRO)                                          INPU 238
   98 IF (NPQ.LT.0) NPGAS = NPQ                                          INPU 239
      NTME = 0                                                           INPU 240
      QSOLAR=QAD1(100)                                                   INPU 241
      ETAL  =ETA1                                                        INPU 242
```

```
      ETAT  =FTA2                                                    INPU 243
      RADIUS=RAD1                                                    INPU 244
      NDBUG =NDBUG1                                                  INPU 245
C                                                                    INPU 246
      READ (9,B0C60S)                                                INPU 247
C                                                                    INPU 248
      DELTSV = DELTA                                                 INPU 249
      ICHK = IHANK                                                   INPU 250
      NDBUG1=NDBUG                                                   INPU 251
      IF (NPGAS.GE.0)                                                INPU 252
     INPGAS =NPQ                                                     INPU 253
      RAD1  =RADIUS                                                  INPU 254
      FTA2  =FTAT                                                    INPU 255
      ETA1  =ETAL                                                    INPU 256
      QAD1(100)=QSOLAR                                               INPU 257
      TWNB = TWALL                                                   INPU 258
      IF ((TWNB.LE.0.).AND.(NDSB.GT.0)) TWNB = TABL                  INPU 259
      DO 100  JMK=1,100                                              INPU 260
      VAPRES(JMK) = STAGEN(JMK)                                      INPU 261
      TIME1 (JMK) = TIME (JMK)                                       INPU 262
      ALT1  (JMK) = ALT  (JMK)                                       INPU 263
      VINF1 (JMK) = VINF (JMK)                                       INPU 264
      AOFA1 (JMK) = AOFA (JMK)                                       INPU 265
      QCONS1(JMK) = QCONS(JMK)                                       INPU 266
  100 QAD1  (JMK) = QI   (JMK)                                       INPU 267
      IF (VINF(1).EQ.0.)   WRITE(6,1359)                             INPU 268
 1359 FORMAT ('- VINF(1)=0., MAY NOT WORK'///)                       INPU 269
      QAD1(100) = QSOLAR                                             INPU 270
      DO 101  I  =1,40                                               INPU 271
      A(1,I) = A1(I)                                                 INPU 272
      A(2,I) = A2(I)                                                 INPU 273
      A(3,I) = A3(I)                                                 INPU 274
      A(4,I) = A4(I)                                                 INPU 275
      B(1,I) = B1(I)                                                 INPU 276
      B(2,I) = B2(I)                                                 INPU 277
      B(3,I) = B3(I)                                                 INPU 278
      B(4,I) = B4(I)                                                 INPU 279
  101 TEMPER( I ) = TEMP( I )                                        INPU 280
      WRITE (6,9) NDBUG                                              INPU 281
    9 FORMAT (15H0END OF INPUT.   76X I2 )                           INPU 282
CARDS MOVED FROM OUTINT - FOR BETTER VIEW, ALSO NPGAS TEST CHANGED NOV70
      IF (IGO.GT.0)GO TO 55                                          OUTPT 41
C     IF (NPGAS-1) 55, 49, 50                                        OUTPT 42
      IF (NPGAS.GT.1)   GO TO 50
   49 QI(100) = QAD1(100) * 3600.                                    OUTPT 43
      GO TO 55                                                       OUTPT 44
```

```
   50 NPGAS=NTRAJ                                                       OUTPT 45
      DO 51 I=1,NPGAS                                                   OUTPT 46
   51 QI(I) = 3600.*QAD1(I)                                             OUTPT 47
   55 CONTINUE
C                                                                       INPU 283
      CALL OUTINT (INPCW)                                               INPU 284
C                                                                       INPU 285
                                                                        INPU 286
      IF (NSEG.EQ.1) GO TO 3                                            INPU 287
      IF (KFUNT.LT.5) GO TO 3                                           INPU 288
      DO 260 I=1,KFUNT                                                  INPU 289
      DO 260 J=2,NSEG                                                   INPU 290
      IF (RHOTBL(I,J).LE.0.)  RHOTBL(I,J)=RHOTBL(I,J-1)                 INPU 291
      IF (TEMTBL(I,J).LE.0.)  TEMTBL(I,J)=TEMTBL(I,J-1)                 INPU 292
      IF (EKTBL (I,J).LE.0.)   EKTBL(I,J)= EKTBL(I,J-1)                 INPU 293
  260 IF (CPTBL (I,J).LE.0.)   CPTBL(I,J)= CPTBL(I,J-1)                 INPU 294
C             ENTER TABLES FOR FIRST SEGMENT                            INPU 295
C             PROG. WILL FILL IN FOR YOU UNTILL IT FINDS MORE ENTRYS    INPU 296
C             AT WHICH TIME ALL SUCCEEDING SEGMENTS WILL HAVE THOSE PROPERINPU 297
C         NOTE  TO CHANGE, ZERO OUT THOSE TO BE CHANGED  I.E.           INPU 298
C             CPTBL(2,1)=40*0,CPTBL(3,1)=40*0,CPTBL(1,1)=NEW VALUES     INPU 299
C             WILL  CHANGE ONLY FIRST 3 ROWS (SEGMENTS) IN CP TABLE     INPU 300
    3 IGO = 0                                                           INPU 301
C            INITIALIZE TEMPS.                                          INPU 302
      DO 16 I=1,NSEG                                                    INPU 303
   16 TEMP(I) = TEMPER(I)                                               INPU 304
      IF (START) 4,5,4                                                  INPU 305
C            START TIME, IF OTHER THAN 0.                               INPU 306
    4 ALTUD = START                                                     INPU 307
    5 IF (HV.NE. 0.) GO TO 22                                           INPU 308
      HV= HOABL                                                         INPU 309
   22 IF (QCONS1(NTRAJ).NE.0.) GO TO 7                                  INPU 310
  245 DO 255   I = 1,NTRAJ                                              INPU 311
  255 QCONS1(I) = 1.0                                                   INPU 312
C                                                                       INPU 313
    7 IF (NDBUG.EQ.0) GO TO 102                                         INPU 314
      WRITE (6,BD060S)                                                  INPU 315
  102 IF (NDIM.LT.2) GO TO 15                                           INPU 316
                                                                        INPU 317
      WRITE (6,200)                                                     INPU 318
  200 FORMAT ( '-TWO-DIMENSIONAL OPTION BEING USED ')                   INPU 319
      READ (9,TWODIM)                                                   INPU 320
      WRITE (6,201)     NAROW,NBCOL,NDIM,(AA(I),I=1,NAROW)              INPU 321
      WRITE (6,202)    (BB(I),I=1,NBCOL)                                INPU 322
      WRITE (6,203)   ((CC(I,J),J=1,10   ),I=1,NAROW)                   INPU 323
      WRITE (6,204)   ((T2(I,J), J=1,10   ),I=1,NAROW)                  INPU 324
```

```
  201 FORMAT (  '-NAROW NBCOL NDIM  '/ 3I6/' THICKNESS  Y,AA'/(5E16.8)) INPU 325
  202 FORMAT (  ' WIDTH      X,RB'/(5F16.8))                            INPU 326
  203 FORMAT (  ' COND. CONST.    ,CC '/ (10E12.5))                     INPU 327
  204 FORMAT (  ' TEMPERATURE (R),T2 '/ (10E12.5))                      INPU 328
                                                                        INPU 329
C                                                                       INPU 330
   15 CALL OUTPUT                                                       INPU 331
C                                                                       INPU 332
      NT = 0                                                            11/25/70
   12 IF (INPCW.LE.0)  GO TO 18                                         INPU 333
      IF (INPCW.LT.NTRAJ) INPCW = NTRAJ                                 INPU 334
      WRITE (6,17) INPCW,  (QINT(I),HEDGE(I),PEDGE(I),UEDGE(I),HRECOV(I)INPU 335
     1  ,TIME1(I) , I=1,INPCW )                                         INPU 336
   17 FORMAT ( 48H- COLD WALL INPUTS - AS PER ENTROPY SWALLOWING  /     INPU 337
     1    27H  NO. OF INPUTS = INPCW =  I3 / 7X,4HQINT,11X,5HHEDGE,11X, INPU 338
     2  5HPEDGE,11X,5HUEDGE,10X,6HHRECOV,11X,5HTIME / (6E16.8) )        INPU 339
   18 IF (PSTMAX .LE.0.) GO TO 120                                      INPU 340
      WRITE (6,119) PSTMAX,HVPSTM                                       INPU 341
  119 FORMAT (7X,'PSTMAX',10X,'HVPSTM'/2E16.8)                          INPU 342
  120 RETURN                                                            INPU 343
      END                                                               INPU 344
```

```
      SUBROUTINE KEMP                                                   KMP    2
C                                                                       KMP    4
C - - -    MARCH 27 1964      JOHN KOHLENBERGER        FORTRAN IV       KMP    3
CKEMP                                                                   KMP    1
C     SUBROUTINE KEMP (ICONF,VINF,TINF,RHOINF,STAGI,WALLI,XX,SWEEP,AOFA,KMP    5
C    1QSUBI                                                             KMP    6
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)KMP    7
     1  ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)    KMP    8
     2  ,    QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)   KMP    9
     3  ,    A(4,40),B(4,40)                                            KMP   10
     4  ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)     KMP   11
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI KMP   12
     1  ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                 KMP   13
     2  ,    AA, BB, CC,CP2, DK, T2                                     KMP   14
      COMMON        ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV            KMP   15
     1  ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                       KMP   16
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,           KASE,KFUNT,    NPQKMP   17
     3  ,    NBCOL,NDBUG,NDIM,NOSB,NPGAS,NGI,NPREST,NPU,NSEG,NSEG1,NSEG2 KMP   18
     4  ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  KMP   19
     5  ,    RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLKMP   20
     6  ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   KMP   21
     7  ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE       KMP   22
     8  ,    XX,XDIST                                                   KMP   23
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      KMP   24
      EQUIVALENCE  (NSEG,NAROW)                                         KMP   25
      NAMELIST /KEMPR/  RHOINF,RHONOT,VINF,VORB,STAGI,WALLI,STAGI,BASEI
     1  ,XX,RADIUS,AMINF,QSPH,ICONF,TINF,QSUBI,ANG,AOFS,QCYL
C     AMINF = VINF/(49.01*(SQRT (TINF)))                                KMP   26
      IF (AMINF.LE.1.01)   GO TO 5
      IF (AMINF-3.0)10,20,20                                            KMP   27
C            SIBULKIN EQUN.                                             KMP   28
C           3220. CHANGED TO 174.0 C.B.A.  1/2/66                       KMP   29
    5 STAGI = SHKI + VINF**2/50061.5
C           WHERE SHKI IS AIINF  (FROM THERMO)
      RECI = STAGI
   10 QSPH = 174.0*(SQRT ((RHOINF*VINF)/XX))*(STAGI-WALLI)              KMP   30
      GO TO (300,300,300,100,200),ICONF                                 KMP   31
C            KEMP AND RIDDLE EQUNS.                                     KMP   32
   20 RHONOT = .0023769                                                 KMP   33
      VORB = 26000.                                                     KMP   34
      BASEI = 129.5                                                     KMP   35
      QSPH=24900.*SQRT (RHOINF/RHONOT)*((VINF/VORB)**3.15)*((STAGI-WALLIKMP   36
     1)/(STAGI-BASEI))*3600./SQRT (XX)                                  KMP   37
```

```
      GO TO (300,300,300,100,200),ICONF                                 KMP   38
C           SPHERE                                                      KMP   39
  100 QSUBI = QSPH                                                      KMP   40
      GO TO 300                                                         KMP   41
C           CYLINDER                                                    KMP   42
  200 ANG = (COS (AOFA/57.2958))*(SIN (SWEEP/57.2958))                  KMP   43
      AOES = ATAN (ANG/(SQRT (1.0-(ANG**2))))                           KMP   44
      QCYL = .931*QSPH*((COS (AOES))**1.5)                              KMP   45
      QSUBI = QCYL                                                      KMP   46
      GO TO 300                                                         KMP   47
  300 IF (NDBUG.GE.1) WRITE(6,KEMPR)
      RETURN
      END                                                               KMP   49
```

```
      SUBROUTINE ONED                                                   COND   2
CONED                                                                   COND   1
C     1ST GET CP(I),EK(I) AS FCNS OF TEMP(I) I = NSEGI, NSEG           COND   3
C - - -      APRIL 8 1964      JOHN KOHLENBERGER         FORTRAN IV      COND   4
C                                                                       COND   5
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)COND   6
     1  ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)     COND   7
     2  ,   QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)    COND   8
     3  ,   A(4,40),B(4,40)                                             COND   9
     4  ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)      COND  10
      COMMON      A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI COND  11
     1  ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                  COND  12
     2  ,   AA, BB, CC,CP2, DK, T2                                      COND  13
      COMMON      ALT,AMINF,AOFA,ANGLE, CCNPR, DELTA,DQSAV              COND  14
     1  ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                        COND  15
     2  ,ICHK,ICK,ICONF,IOD,INB4,INB42,IREC,          KASE,KFUNT,    NPQCOND  16
     3  ,   NBCOL,NDBUG,NDIM,NOSB,NPGAS,NCI,NPREST,NPU,NSEG,NSEG1,NSEG2 COND  17
     4  ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCCNS,QINSD,QNET,QSAV,QSUBI   COND  18
     5  ,   RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLCOND  19
     6  ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR    COND  20
     7  ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE        COND  21
     8  ,   XX,XDIST                                                    COND  22
     9      ,ETATP,ETALP,HVPSTM,PSTMX,NT
C
      EQUIVALENCE (NSEG,NAROW)                                          COND  24
C
      DOUBLE PRECISION  TNEW (40),A1,TOLD(40)
C
      DATA  TIMB4/10000./
C
C       11/20/70  DOUBLE PRECISION (12 DIGITS) CONDUCTION ROUTINE
C                       ESPECIALLY FOR SMALL TIME STEPS
   10 CALL GETCPK                                                       COND  26
   11 IF (NSEG-1)12,12,14                                               COND  27
   12 TEMP(1) = (DELTA/(RHO(1)*EL(1)*CP(1)*25.0))*(QNET/12.0+HINSD*     COND  28
     1 (TINSD-TEMP(1))/12.0)+TEMP(1)                                    COND  29
      GO TO 63                                                          COND  30
   14 DO 50 I=NSEG1,NSEG                                                COND  31
   15 A1 = DELTA/(RHO(I)*EL(I)*CP(I)*25.0)                              COND  32
      IF (I-NSEG2)25,25,30                                              COND  33
   25 IF ( NT .GT.  1  )  GO TO 27
CARDS BELOW ARE FOR FIRST TIME THROUGH
      DO 26  J= NSEG1,NSEG
```

```
      TOLD(J) = TEMP(J)
   26 CONTINUE
C     NS1B4 =  NSEG1
C     NSEGB4= NSEG
C     NS2B4 = NSEG2
      TIMB4 = TIME
   27 IF (NSEG2.EQ.NSEG1) GO TO 28
      TOLD(NSEG2) = TEMP(NSEG2)
CARD ABOVE,USED IN CASE OF ABLATION
   28 TNEW(I) = A1*(QNET/12.0+(TOLD(I+1)-TOLD(I))/(.5*(FL(I)/EK(I) +
     1FL(I+1)/ FK(I+1)))) + TOLD(I)
      GO TO 50                                                          COND  36
   30 IF (I-NSEG) 35,40,40                                              COND  37
   35 TNEW(I) = A1*((TOLD(I-1)-TEMP(I))/(.5*(EL(I-1)/EK(I-1)+FL(I)
     1 /EK(I))) + (TOLD(I+1)-TOLD(I))/(.5*(FL(I+1)/EK(I+1)+EL(I+1)
     2 /EK(I+1)))) + TOLD(I)
      GO TO 50                                                          COND  41
   40 TNEW(I) = A1*((TOLD(I-1)-TOLD(I))/(.5*(EL(I-1)/EK(I-1)+FL(I)
     1/EK(I))) + HINSD*(TINSD-TOLD(NSEG))/12.0) + TOLD(I)
   50 CONTINUE                                                          COND  44
      DO 60 I = NSEG1,NSEG                                              COND  45
      IF (NDBUG.NE.10)   GO TO 59
      WRITE(6,1970) TIMB4,TOLD
      WRITE(6,1970) TIME,TNEW
 1970 FORMAT(' DEBEG -UNED-, TIME AND TEMPERATURE',D16.8/(8D16.8))
   59 CONTINUE
      TOLD(I) = TNEW(I)
   60 TEMP(I) = TNEW(I)                                                 COND  46
   63 DQINSD = HINSD*(TEMP(NSEG)-TINSD)*DELTA/3600.                     COND  47
      QINSD = QINSD+DQINSD                                              COND  48
   65 TWALL = TEMP(NSEG2)                                               COND  49
      IF (NDBUG.NE.10)   GO TO 69
      WRITE (6,1971) NSEG,NSEG1,NSEG2,A1,CNET,HINSD,TINSD,QINSD,TWALL
 1971 FORMAT ('   NSEG NSEG1 NSEG2',14X,'A1',12X,'QNET'/3I6,2E16.8/
     111X,'HINSD',11X,'TINSD',11X,'QINSD',11X,'TWALL'/4E16.8)
   69 TIMB4 = TIME
   70 RETURN                                                            COND  50
      END                                                               COND  51
```

```
      SUBROUTINE OUTINT  (INPCW)
COUTINT   INTEGERS FOR  BD060S         AERODYNAMIC HEATING PROGRAM  1968OUTPT   1
C - - -     JANUARY 17,1968   JOHN KOHLENBERGER          FORTRAN IV       OUTPT   2
      DIMENSION   TIME1(100),ALTI(100),VINF1(100),AOFA1(100),HINSD1(100)OUTPT   4
     1  ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)     OUTPT   5
     2  ,   QI(100),QAD1(100), CP(40),FK(40),EL(40),RHO(40),TEMP(40)    OUTPT   6
     3  ,   A(4,40),B(4,40)                                             OUTPT   7
     4  ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)      OUTPT   8
      COMMON      A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI OUTPT   9
     1  ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                  OUTPT  10
     2  ,   AA, BB, CC,CP2, DK, T2                                      OUTPT  11
      COMMON      ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV              OUTPT  12
     1  ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                        OUTPT  13
     2  ,ICHK,ICK,ICONF,IGO,INR4,INB42,IRFC,          KASE,KFUNT,    NPQOUTPT  14
     3  ,   NBCOL,NDBUG,NDIM,NOSB,NPGAS,NCI,NPREST,NPU,NSEG,NSEG1,NSEG2 OUTPT  15
     4  ,   NTPAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   OUTPT  16
     5  ,   RADIUS,RFCI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLOUTPT  17
     6  ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR    OUTPT  18
     7  ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABOOT,WALLI,WRITE        OUTPT  19
     8  ,   XX,XDIST                                                    OUTPT  20
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      OUTPT  21
      COMMON / MIKE / HV2,ETAL2,ETAT2,XI, FV1P(10),FV1S(10),FV2P(10),   OUTPT  22
     *      FV2S(10),STGTBL(10),SHRTBL(10),NOPSTG,NOPSHR                OUTPT  23
     2   ,  QSONIC,SHEAR,HOABLM,HF, ISTETS,  NOCHEK,                    OUTPT  24
     3      RHOTBL(10,40),CPTBL(10,40),EKTBL(10,40),TEMTBL(10,40)       OUTPT  25
      COMMON /BLOK1/ TITLE(20)
      COMMON /WERNER/   TRVRSL,TFB(100),RHOFB(100),PFB(100),ALTCM(100)
      DIMENSION WORD(24)                                                OUTPT  29
      DATA WORD/'PLAT','WEDG','CONE','SPHR','CYLN','BLNT',18*'    ' /
      EQUIVALENCE (NSEG,NAROW)                                          OUTPT  28
      WRITE (6,6) TITLE                                                 OUTPT  32
    6 FORMAT (1H1,20A4 )                                                OUTPT  33
   55 WRITE (6,56) KASE                                                 OUTPT  48
   56 FORMAT (31H IDENTIFICATION = &BD060S        /    5H KASE I4   )   OUTPT  49
      IF (NOSB)    60,60,58                                             OUTPT  50
   58 CALL PRINTH (WORD(6))                                             OUTPT  51
   60 CALL PRINTH (WORD(ICONF))                                         OUTPT  52
      IF (ALT) 300,300,301                                              OUTPT  53
C          ALT = START , INITIALLY                                      OUTPT  54
  300 TIME = TIME1(1)                                                   OUTPT  55
      GO TO 105                                                         OUTPT  56
  301 TIME = ALT                                                        OUTPT  57
      WRITE (6,302) ALT                                                 OUTPT  58
```

```
  302 FORMAT (9H0 START =  E20.8 ///)                                   OUTPT 59
  105 WRITE (6,106)      ICONF,ICHK,ICHK, INPCW,ISTETS,KASE,KFUNT,NBCOL,OUTIT 60
    1 NOBUG,NDIM, NOPSHR,NOPSTG,      NOSB,NPGAS,NPQ,NPU,NQI,NSEG,NTRAJ OUTIT 61
    2 ,NOCHEK,     IGO                                                  OUTIT061
  106 FORMAT ( 9H INTEGERS /          5X,5HICONF,6X,4HICHK,5X,5HIHANK,  OUTIT 62
    1   5X,5HINPCW,4X,6HISTETS,6X,4HKASE,5X,5HKFUNT,5X,5HNBCOL,5X,      OUTIT 63
    2   5HNOBUG,6X,4HNDIM /10I10/ 4X,6HNOPSHR,4X,6HNOPSTG,6X,4HNOSB,    OUTIT 64
    3   5X,5HNPGAS,7X,3HNPQ,7X,3HNPU,7X,3HNQI,6X,4HNSEG,5X,5HNTRAJ      OUTIT 65
    4 ,4X,6HNOCHEK/10I10/  I10)                                               66
C      ICONF = 1 FLAT PLATE                                             OUTPT 65
C            = 2 WEDGE                                                  OUTPT 66
C            = 3 CONE                                                   OUTPT 67
C            = 4 SPHERE                                                 OUTPT 68
C            = 5 CYLINDER                                               OUTPT 69
C      NDIM  = NO. OF DIMENSIONS ( 1 OR 2 )                             OUTPT 70
C      NTRAJ = NO. OF TRAJECTORY CARDS (MAXIMUM OF 100 )                OUTPT 71
C      KFUNT = 1 EK(I) AND CP(I) ARE BOTH FCNS OF TEMPERATURE           OUTPT 72
C            = 2 EK(I) IS A FCN OF TEMPERATURE, CP(I) IS CONSTANT       OUTPT 73
C            = 3 EK(I) IS CONSTANT, CP(I) IS A FCN OF TEMPERATURE       OUTPT 74
C            = 4 BOTH ARE CONSTANT                                      OUTPT 75
C      NSEG  =NARCW = NUMBER OF SEGMENTS(MAXIMUM OF 20 IN 1D, 10 IN 2D ) OUTPT 76
C      NBCOL = IN 2 DIMENSIONS IS NO. OFCOLUMS (MAX. PRESENT FOR ANY GIVEOUTPT 77
C               N SLAB, NOT TO EXCEED 10)                               OUTPT 78
       RETURN                                                           OUTIT 99
       END
```

```
      SUBROUTINE OUTPUT                                                 OUTPT   3
COUTPUT   ROUTINE  FOR  BD06E          AERODYNAMIC HEATING PROGRAM  1967OUTPT   1
C - - -     OCTOBER 14 1970   JOHN KOHLENBERGER        FORTRAN IV     OUTP    2
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)OUTPT   4
     1  ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)     OUTPT   5
     2  ,   QI(100),QAD1(100), CP(40),FK(40),EL(40),RHO(40),TEMP(40)    OUTPT   6
     3  ,   A(4,40),B(4,40)                                             OUTPT   7
     4  ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)      OUTPT   8
C
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI OUTPT   9
     1  ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                  OUTPT  10
     2  ,   AA, BB, CC,CP2, DK, T2                                      OUTPT  11
      COMMON       ALT,AMINF,AOFA,ANGLE, CCNPR, DELTA, DQSAV            OUTPT  12
     1  ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                        OUTPT  13
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,          KASE,KFUNT,    NPQOUTPT  14
     3  ,   NBCOL,NDBUG,NDIM,NOSB,NPGAS,NCI,NPREST,NPU,NSEG,NSEG1,NSEG2 OUTPT  15
     4  ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   OUTPT  16
     5  ,   RADIUS,RECI,REFI,  FMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLOUTPT  17
     6  ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR    OUTPT  18
     7  ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE        OUTPT  19
     8  ,   XX,XDIST                                                    OUTPT  20
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      OUTPT  21
      COMMON / MIKE / HV2,ETAL2,ETAT2,XI, FV1P(10),FV1S(10),FV2P(10),   OUTPT  22
     *       FV2S(10),STGTBL(10),SHRTBL(10),NOPSTG,NOPSHR               OUTPT  23
     2    ,  QSONIC,SHEAR,HOABLM,HF, ISTETS,  NOCHEK,                   OUTPT  24
     3       RHOTBL(10,40),CPTBL(10,40),EKTBL(10,40),TEMTBL(10,40)      OUTPT  25
      COMMON /BLOK1/ TITLE(20)
      COMMON /WERNER/   TRVRSL,TFB(100),RHOFB(100),PFB(100),ALTCM(100)
      COMMON /PHIL/ NPQS,QSTAR(10),PSTAGN(10)
      COMMON /RALPH/ TWTAB(20),EMTAB(20),NEMIS,NQCOR,QABSF(20),TWA(20)
C
      EQUIVALENCE (NSEG,NAROW)                                          OUTPT  28
      EQUIVALENCE (QCALT,TRANLT)
C
      WRITE (6,6) TITLE                                                 OUTPT  32
    6 FORMAT (1H1,20A4 )                                                OUTPT  33
    1 FORMAT( 14I5 )                                                    OUTPT  34
    2 FORMAT( 50F1.0 )                                                  OUTPT  35
    4 FORMAT(4E14.0)                                                    OUTPT  36
    5 FORMAT( 5E14 0)                                                   OUTPT  37
    7 FORMAT( 7E10.0)                                                   OUTPT  38
    8 FORMAT (10F7.0  )                                                 OUTPT  39
   12 FORMAT ( 12A6 )                                                   OUTPT  40
```

```
  125 WRITE (6,126) TRANR,TRANLT,ANGLE,SWEEP,XDIST,CONPR                   OUTPT 79
  126 FORMAT (10CHO     TRANR               TRANLT               ANGLE     OUTPT 80
     1 SWEEP            XDIST               CONPR              /6E16.8)    OUTPT 81
C       TRANR = TRANSITION REYNOLDS NO.                                    OUTPT 82
C       ANGLE = WEDGE OR CONE  HALF ANGLE                                  OUTPT 83
C       SWEEP = SWEEP ANGLE                                                OUTPT 84
C       XDIST = DISTANCE FROM LEADING EDGE FOR FLAT PLATE, WEDGE OR CONE  OUTPT 85
C             = DIAMETER FOR SPHERE OR CYLINDER                    (FEET) OUTPT 86
C       EMIS  = SURFACE EMISSIVITY                                         OUTPT 87
  145 WRITE (6,146) DELTA,WRITE,STOP,TABL,HV,ETA1,ETA2,EMIS                OUTPT 88
  146 FORMAT (1HC7X,5HDELTA,15X,5HWRITE,15X,4HSTOP/3E20.8///               OUTPT 89
     1 7X,5H TABL,15X,2HHV,18X,4HETAL,16X,4HETAT,15X,5HEMIS   / 5E20.8)    OUTPT 90
      IF (ISTETS.GT.0) WRITE (6,1775)                                      OUTPT 91
 1775 FORMAT (31H-WIYH STETSON LAMINAR ABLATION    )                       OUTPT 92
C       ETA1,ETA2 = USED IN HOABL EQUATION  -SEE HV-                       OUTPT 93
C       DELTA = TIME STEP SIZE        ( SECONDS )                          OUTPT 94
C       WRITE = PRINT OUT TIME INTERVAL      ( SECONDS )                   OUTPT 95
C       STOP  = END OF FLITE TIME                                          OUTPT 96
C       TABL = ABLATION TEMPERATURE ( DEGREES R )                          OUTPT 97
C       HV    = USED IN HOABL EQUN. (HOABL=HV+ETA*(RECI-WALLI))            OUTPT 98
C       CONPR = CONE PRESSURE  AS USED IN *PME*          10/23/63          OUTPT 99
      IF (KFUNT.GT.4) GO TO 2665                                           OUTPT100
      IF (KFUNT.GT.0)   GO TO 159
  151 FORMAT ('    KFUNT = 0,    DK AND CP2 LISTED BELOW')
      WRITE (6,151)
      DO 155 I=1,NAROW
      WRITE (6,152) I, (DK(I,K), K=1,NBCOL)
  152 FORMAT(' ROW',I6,  ' DK'/10E13.5)
      WRITE (6,153)   (CP2(I,K), K=1,NBCOL)
  153 FORMAT(9X          ,'CP2'/10E13.5)
      WRITE(6,154)  RHO(I)
  154 FORMAT (9X,'RHO=',E13.5)
  155 CONTINUE
      GO TO 260
  159 WRITE (6,217)  (J,A(1,J),A(2,J),A(3,J),A(4,J) ,J=1,NSEG)             OUTPT101
  217 FORMAT (3HC J,10X, 5HA1(J),11X,5HA2(J),11X,5HA3(J),11X,5HA4(J)       OUTPT102
     1  / (I3,4E16.8) )                                                    OUTPT103
      WRITE (6,227)  (J,B(1,J),B(2,J),B(3,J),B(4,J) ,J=1,NSEG)             OUTPT104
  227 FORMAT (3HC J,10X, 5HB1(J),11X,5HB2(J),11X,5HB3(J),11X,5HB4(J)       OUTPT105
     1  / (I3,4E16.8)  )                                                   OUTPT106
C       A(I,J) = COEFFICIENTS OF JTH SEGMENT FOR COMPUTING EK(J) THUSLY    OUTPT107
C                EK(J) = A(1,J) + A(2,J)*T + A(3,J)*T**2 + A(4,J)*T**3     OUTPT108
C       B(I,J) = ARE TO CP(J)  AS THE A S ARE TO EK(J)                     OUTPT109
  165 WRITE (6,166) NSEG,(RHO(I),I=1,NSEG)                                 OUTPT110
  166 FORMAT ( 26H0               RHO(I), I=1,  I2 /  (5E20.8 )          ) OUTPT111
C       RHO(I) = SPECIFIC WT. OF SEGMENT I      ( LBS/FT**3  )             OUTPT112
```

```
      GO TO 260                                                          OUTPT113
 2665 DO 2267 I=1,NSEG                                                   OUTPT114
 2267 WRITE (6,2266) I,(TEMTBL(J,I),EKTBL(J,I),CPTBL(J,I),RHOTBL(J,I),   OUTPT115
     1        J=1,KFUNT )                                                OUTPT116
 2266 FORMAT (  9H-SEGMENT I3/    9H   TEMTBL,8X, 5HEKTBL,11X, 5HCPTBL,  OUTPT117
     1  11X,6HRHOTBL / (4E16.8))                                         OUTPT118
  260 GO TO ( 275, 305 ), NDIM                                           OUTPT119
C      1 DIM INPUT                                                       OUTPT120
  275 WRITE (6,276) NSEG,(FL(I),I=1,NSEG)                                OUTPT121
  276 FORMAT ( 22HO          FL(I), I=1,  I2 /  (5E20.8 )     )          OUTPT122
C      FL(I) = THICKNESS OF ITH SEGMENT    ( INCHES )                    OUTPT123
  295 WRITE (6,296) NSEG,(TEMP(I),I=1,NSEG)                              OUTPT124
  296 FORMAT ( 22HO        TEMP(I), I=1,  I2 /  (5F20.8 )     )          OUTPT125
C      TEMP(I) = TEMPERATURE OF ITH SEGMENT        ( DEGREES R )         OUTPT126
      GO TO 400                                                          OUTPT127
C      2 DIM INPUT                                                       OUTPT128
C      AA(I) = THICKNESS OF ITH SEGMENT IN 2D. (SAME AS FL(I) IN 1D. )IN.OUTPT129
C      BB(I) = WIDTH OF ITH SEGMENT IN 2D.                    (INCHES)   OUTPT130
  305 WRITE (6,306) NAROW,(AA(I),I=1,NAROW)                              OUTPT131
  306 FORMAT(1HOI3,14H VALUES OF AA / (5E20.8))                          OUTPT132
  325 WRITE (6,326) NBCOL,(BB(J),           J=1,NBCOL)                   OUTPT133
  326 FORMAT(1HOI3,14H VALUES OF BB / (5E20.8))                          OUTPT134
      WRITE (6,327) (( I J,CC(I,J),T2(I,J),J=1,NBCOL),I=1,NAROW)         OUTPT135
  327 FORMAT (8HO I , J 10X, 8H CC(I,J) 10X, 8H T2(I,J)/ (  2I3,5X,      OUTPT136
     1 2E18.8))                                                          OUTPT137
  400 WRITE (6,237)  (I,TIME1(I),ALT1(I),VINF1(I),AOFA1(I),HINSD1(I),    OUTPT138
     1 TINSD1(I),QCONS1(I),QI(I),TEMPW(I),        VAPRES(I) , I=1,NTRAJ) OUTPT139
C    1 TINSD1(I),QCONS1(I),QI(I),TEMPW(I),CPG(I),VAPRES(I) , I=1,NTRAJ)  OUTPT139
  237 FORMAT (3H- I 4X,4HTIME 4X,3HALT 6X,4HVINF 4X,4HAOFA 2X,5HHINSD    OUTPT140
C    1 2X,5HTINSD 2X,5HQCONS 2X,7HQI*3600 5X,5HTEMPW 2X,4X,3HCPG 5X,     OUTPT141
     1 2X,5HTINSD 2X,5HQCONS 2X,7HQI*3600       ,13X,5HTEMPW,6X          OUTPT141
     2 6HVAPRES   /  ( 1X,I2,2X,F7.1,1X,F8.1,1X,F8.1,                    OUTPT142
C    2 2X,F4.1,2X,F5.1,2X,F5.1,2X,F5.1,2X,E10.3 ,2X,F5.1,2X,F8.4,2X,F5.1OUTPT143
     2 2X,F4.1,2X,F5.1,2X,F5.1,2X,F5.1,2X,E10.3 ,9X,       F8.4,2X,F5.1OUTPT143
     3 ) )                                                               OUTPT144
C      TRAJECTORY INPUT                                                  OUTPT145
C      TIME =  TIME ( SECONDS )                                          OUTPT146
C      ALT  =  ALTITUDE ( FEET )                                         OUTPT147
C      VINF =  VELOCITY (FT/SEC )                                        OUTPT148
C      AOFA =  ANGLE OF ATTACK ( DEGREES )                               OUTPT149
C      HINSD = INSIDE HEAT TRANSFER  ( BTU/HR FT**2   DEGREE R  )        OUTPT150
C      TINSD = INSIDE TEMPERATURE ( DEGREES R )                          OUTPT151
C      IF THE QCONS(I) ARE NOT READ IN THEY SHOULD ALL = 1               OUTPT152
      NTRAV = TRVRSL
      IF (TRVRSL.GT.0.)        WRITE(6,398) TRVRSL,(I,ALTCM(I),TFB(I),   OUTP
     1        RHOFB(I),PFB(I),I=1,NTRAV)                                 OUTPT
```

```
  399 FORMAT (1H-,' TRVRSL = 'F6.1 /  8X, 'ALTCM',13X,'TFB',13X,'RHOFB',OUTP
     1 11X,'PFB'  / (I3,4E16.8) )                                         OUTPT
      IF(NOSB) 401,404,401                                              OUTPT153
  401 TABL = TWNB                                                       OUTPT154
      WRITE (6,402) NOSB,RADIUS,ANGLE,TWNB                              OUTPT155
  402 FORMAT (35HONOSE BLUNTING ROUTINE WILL BE USED 7H ,NOSB= I3 //    OUTPT156
     17X,7H RADIUS 14X,6HANGLE 14X,5HTWALL / 3E20.8)                    OUTPT157
      IF (PSTMAX.LE.C.) GO TO 404
      IF (HF.LT.1.E-8)   HF = 0.0
      WRITE (6,406)   PSTMAX,HVPSTM,HOABLM
  406 FORMAT (' PSTMAX',10X,'HVPSTM',10X,'HOABLM'/3E16.8)
      WRITE (6,1776)    HF                                              OUTPT170
 1776 FORMAT (6H HF = E16.8)                                            OUTPT171
      PRCNT = 1.-XI                                                     OUTPT159
      NOP = MAX0 (NOPSHR,NOPSTG)                                        OUTPT160
      WRITE (6,403)          HV,ETA1,ETA2,XI                            OUTPT161
  403 FORMAT ( 18H  MATERIAL NO. 1  / 7X,2HHV,14X,4HETAL,12X,4HETAT,12X OUTPT162
     * ,2HX!/     4E16.8)                                               OUTPT163
      IF (XI.EQ.1.) GO TO 404                                           OUTPT158
      WRITE (6,405) HV2,ETAL2,ETAT2,PRCNT,NOPSTG,NOPSHR,                OUTPT164
     *(FV1P(I),FV1S(I),STGTBL(I)     ,SHRTBL(I),FV2P(I),FV2S(I), I=1,NOP)OUTPT165
  405 FORMAT(  18H  MATERIAL NO. 2  /7X,3HHV2,14X,5HETAL2,11X,5HETAT2,  OUTPT166
     1 11X,5H1.-XI / 4E16.8 /32X,9H NOPSTG = I3, 4X,9H NOPSHR = I3 /    OUTPT167
     2 7X,5H FV1P,11X, 4HFV1S,12X,6HSTGTBL,10X,6HSHRTBL,10X,4HFV2P,12X  OUTPT168
     3 ,4HFV2S     /(6E16.8))                                           OUTPT169
  404 IF (NPGAS)  410,421,421                                           OUTPT172
  410 NOP = IABS (NPGAS)                                                OUTPT173
      WRITE (6,413) NOP,(VAPRES(I),TEMPW(I),I=1,NOP)                    OUTPT174
  413 FORMAT (44H-STAGNATION ENTHALPY VS. HOABL, NO. OF PTS.= I3/       OUTPT175
     1 (2E20.8))                                                        OUTPT176
      GO TO 424                                                         OUTPT177
  421 WRITE (6,422) QAD1(100)                                           OUTPT178
  422 FORMAT (7X,7H QSOLAR / E20.8)                                     OUTPT179
  424 IF (NPQS.LE.0)   GO TO 430
      WRITE (6,426) NPQS,(QSTAR(I),PSTAGN(I),I=1,NPQS)
  426 FORMAT (1H- ' QSTAR VS. PSTAGN - TABLE TO COMPUTE HV, HAS',I6,
     1' VALUES (N .S)'/(2E13.5))
  430 IF (NEMIS.LE.0) GO TO 440
      WRITE (6,428) (TWTAB(I),EMTAB(I),I=1,NEMIS)
  428 FORMAT ('- VARIABLE EMISSIVITY, AS A FUNCTION OF TWALL, BEING USED
     1'/7X,'TWTAB',10X,'EMTAB',10X,'TWTAB',10X,'EMTAB',10X,'TWTAB',10X,
     2  'EMTAB'/(6E16.8))
  440 IF (NQCOR.LE.0) GO TO 500
      WRITE (6,431)  (TWA(I),QABSF(I),I=1,NQCOR)
  431 FORMAT ('- QCONS WILL BE MULTIPLIED BY QABSF AT TWA'/ ' TWA (TWALL
     1)      QABSF (Q FACTOR)' / (2E16.8))
```

```
  500 CONTINUE
      WRITE (6,425) NDBUG                                                   OUTPT180
  425 FORMAT (22H0END OF DATA PRINTOUT  50X,6HNDBUG=I2/1H1 )                OUTPT181
      RETURN                                                                OUTPT182
      END                                                                   OUTPT183
```

```
      SUBROUTINE PME                                                    PRAND  5
C PME  -   PRANDLT MAYER EXPANSION        TEMP.,PRESSURE,VELOCITY RATIOSPRAND  1
C - - -    APRIL 22 1964      JOHN KOHLENBERGER          FORTRAN IV     PRAND  2
C     SUBROUTINE PME (TINF,VINF,AOFA,RHOINF,PINF,IREC,RECI,WALLI,REFRHO,PRAND  3
C    1VSHK,REFMU)                                                       PRAND  4
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)PRAND  6
     1  ,     TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)   PRAND  7
     2  ,     QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)  PRAND  8
     3  ,     A(4,40),B(4,40)                                           PRAND  9
     4  ,     AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)    PRAND 10
      COMMON        A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI PRAND 11
     1  ,     RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                PRAND 12
     2  ,     AA, BB, CC,CP2, DK, T2                                    PRAND 13
      COMMON        ALT,AMINF,AOFA,ANGLE, CCNPR, DELTA,DQSAV            PRAND 14
     1  ,     EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                      PRAND 15
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,           KASE KFUNT,    NPQPRAND 16
     3  ,     NBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 PRAND 17
     4  ,     NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  PRAND 18
     5  ,     RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLPRAND 19
     6  ,     SHKI,SHKMU, STOP,SWEEP,STAGI, TA3L,TIMF,TINF,TINSD,TRANR  PRAND 20
     7  ,     TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE      PRAND 21
     8  ,     XX,XDIST                                                  PRAND 22
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      PRAND 23
      EQUIVALENCE (NSEG,NAROW),(REFRM,REFRN)                            PRAND 24
      TRUANG = ANGLE + AOFA                                             PRAND 25
      AMINF = VINF/(49.01*SQRT (TINF))                                  PRAND 26
      IF (AMINF - 1.0)20,20,10                                          PRAND 27
   10 BETAI = SQRT  (AMINF*AMINF-1.0)                                   PRAND 28
      UNI = 140.34601*ATAN (.40825*BETAI)-57.29578*ATAN (BETAI)         PRAND 29
      IF (TRUANG+.05) 30,30,20                                          PRAND 30
      PR = 1.0                                                          PRAND 31
      TR = 1.0                                                          PRAND 32
      VR = 1.0                                                          PRAND 33
      RM = 1.0                                                          PRAND 34
      DR = 1.0                                                          PRAND 35
      GO TO 110                                                         PRAND 36
   30 UNE = UNI - TRUANG                                                PRAND 37
      IF (UNE-50.)40,40,50                                              PRAND 38
   40 AME = -26.301065/(UNE-81.77828)+.78959002+.02791663*UNE           PRAND 39
      GO TO 90                                                          PRAND 40
   50 IF (UNE - 102.32)60,60,70                                         PRAND 41
   60 AME = -280.09435/(UNE-130.19996)-.89411241+8.2845071E-03*UNE      PRAND 42
      GO TO 90                                                          PRAND 43
```

```
   70 AME = -284.84684/(UNE-130.43433)-1.C199074+8.6803698E-03*UNE       PRAND 44
   90 PR = ((2.+.4*AMINF*AMINF)/(2.+.4*AME*AME))**3.5                     PRAND 45
      IF (CONPR .GT. 0.) GO TO 94                                         PRAND 46
      COMPR = 1.E-3                                                       PRAND 47
C           COMPUTED PR  (NOT TO OVER RIDE INPUT VALUE WHEN PR IS SMALL)PRAND 48
      IF (PR .GE. COMPR) GO TO 100                                        PRAND 49
      PR = COMPR                                                          PRAND 50
      GO TO 96                                                            PRAND 51
   94 IF (PR .GE. CONPR) GO TO 100                                        PRAND 52
      PR = CONPR                                                          PRAND 53
   96 AME = SQRT( (2.+4.*AMINF**2)/(.4*PR**.2857143) -5.  )               PRAND 54
  100 TR = (2.+.4*AMINF*AMINF)/(2.+.4*AME*AME)                            PRAND 55
      VR = (AME/AMINF)*SQRT (TR)                                          PRAND 56
      DR = PR/TR                                                          PRAND 57
  110 PSHK = PR*PINF                                                      PRAND 58
      TSHK = TR*TINF                                                      PRAND 59
      RHOSHK = DR*RHOINF                                                  PRAND 60
      VSHK = VR*VINF                                                      PRAND 61
      SHKI = 2.2864+.2272*TSHK+.11571E-4*(TSHK**2)+.23676E-9*(TSHK**3)  PRAND 62
  119 PR = .76                                                            PRAND 63
      GO TO (120,130,180),IREC                                            PRAND 64
  120 REC = SQRT(PR)                                                      PRAND 65
      GO TO 140                                                           PRAND 66
  130 REC = PR**.333333                                                   PRAND 67
  140 RECI = SHKI+(REC*(VSHK**2))/50061.5                                 PRAND 68
      WALLI=2.2864+.2272*TWALL+.11571E-4*(TWALL**2)+.23676E-9*(TWALL**3)PRAND 69
      REFI = SHKI+0.5*(WALLI-SHKI)+.22*(RECI-SHKI)                        PRAND 70
      IF (REFI - 1300.)150,160,160                                        PRAND 71
  150 REFMU = .42642E-7*(REFI**.493)                                      PRAND 72
      REFRHO = .576E-4*(REFI**(-.849))*PSHK                               PRAND 73
      GO TO 170                                                           PRAND 74
  160 REFMU = .28428E-6*(REFI**.228)                                      PRAND 75
      REFRHO = .865E-5*(REFI**(-.584))*PSHK                               PRAND 76
  170 REFRM = .2465E-11*(REFI**(-.356))*PSHK                              PRAND 77
      GO TO 200                                                           PRAND 78
  180 IF (SHKI-1300.)185,190,190                                          PRAND 79
  185 SHKMU = .42642E-7*(SHKI**.493)                                      PRAND 80
      GO TO 200                                                           PRAND 81
  190 SHKMU = .28428E-6*(SHKI**.228)                                      PRAND 82
C *************** CHANGES ******************                             PRAND 83
  200 IF (NDBUG-3) 210,117,210                                            PRAND 84
  117 WRITE (6,118) PR,CONPR,PSHK,DR,AME,AMINF,                           PRAND 85
     1        TRUANG,UNE,REFRM,RHOSHK,IREC                                PRAND 86
  118 FORMAT (20H .PME.  DEBUG PRINT /7X,27H PR,CONPR,PSHK,DR,AME,AMINF/PRAND 87
     1        6E18.8/7X,45H TRUANG(ANGLE-ACFA), UNE, REFRM, RHOSHK, IREC/PRAND 88
     2        4E18.8,I8 )                                                 PRAND 89
```

```
C **************** CHANGES ******************        PRAND 90
  210 RETURN                                         PRAND 91
      END                                            PRAND 92
```

```
      SUBROUTINE PRINT                                                  PRNT   1
C - - -    OCTOBER 14 1970    JOHN KOHLENBERGER         FORTRAN IV      PRNT   2
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)PRNT   3
     1  ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)    PRNT   4
     2  ,    QI(100),QADI(100), CP(40),FK(40),EL(40),RHO(40),TEMP(40)   PRNT   5
     3  ,    A(4,40),B(4,40)                                            PRNT   6
     4  ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)     PRNT   7
      COMMON       A,ALT1,AOFA1,B, CP,CPG, FK,EL, HINSD1, QADI,QCONS1,QI PRNT   8
     1  ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                 PRNT   9
     2  ,    AA, BB, CC,CP2, DK, T2                                     PRNT  10
      COMMON       ALT,AMINF,AOFA,ANGLE, CCNPR, DELTA,DQSAV             PRNT  11
     1  ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                       PRNT  12
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,           KASE,KFUNT,    NPQPRNT  13
     3  ,    NBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 PRNT  14
     4  ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  PRNT  15
     5  ,    RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLPRNT  16
     6  ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   PRNT  17
     7  ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE       PRNT  18
     8  ,    XX,XDIST                                                   PRNT  19
     9     ,ETATP,ETALP,HVPSTM,PSTMX,NT
      COMMON /CHARAB/ NTME,TSTOR(50),QO(50),PO(50),HSTAB(50),HLTAB(50), PRNT  21
     1 UETAB(50)                                                        PRNT  22
      COMMON / SWALOW/  QINT(50),HEDGE(50),PEDGE(50),UEDGE(50),HRECOV(50PRNT  23
     1)  ,   INPCW                                                      PRNT  24
C            FOR ARCHIF OSSIN - COLD WALL INPUTS FROM ENTROPY SWALLOWING PRNT  25
      COMMON / MIKE / HV2,ETAL2,ETAT2,XI, FV1P(10),FV1S(10),FV2P(10),   PRNT  26
     *       FV2S(10),STGTBL(10),SHRTBL(10),NOPSTG,NOPSHR               PRNT  27
     *  ,  QSONIC,SHEAR,HOABLM                                          PRNT  28
      EQUIVALENCE  (NSEG,NAROW)                                         PRNT  29
      NAMELIST /COMON/
     1             ALT,AMINF,AOFA,ANGLE, CCNPR, DELTA,DQSAV
     1  ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,           KASE,KFUNT,    NPQPRNT
     3  ,    NBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 PRNT
     4  ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   P
     5  ,    RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLPRNT
     6  ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR
     7  ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE
     8  ,    XX,XDIST
     9  ,ETATP,ETALP,HVPSTM,PSTMAX
      Q3600 = QSUBI/3600.                                               PRNT  30
      H3600 = HSUBI/3600.                                               PRNT  31
      QN3600 = QNET/3600.                                               PRNT  32
```

```
   20 IF (NT.GE.1)   GO TO 40
   30 QNSUM = 0.0                                                   PRNT  34
      QN = 0.0                                                      PRNT  35
      TIM1 = TIME1(1)                                               PRNT  36
      TIM2 = TIME                                                   PRNT  37
      GO TO 45                                                      PRNT  38
   40 TIM1 = TIM2                                                   PRNT  39
      TIM2 = TIME                                                   PRNT  40
   45 IF(QN3600)  60,50,50                                          PRNT  41
   50 QNSUM = QNSUM + (QN3600+QN) /2. * (TIM2- TIM1)                PRNT  42
   60 QN = QN3600                                                   PRNT  43
      DYNP = RHOINF * VINF*VINF/2.                                  PRNT  44
      IF (RECI.NE.WALLI) GO TO 64
      TIM1 = TIME-DELTA
      GO TO 65
   64 SHEARS=.025 *QSUBI*VSHK /((RECI-WALLI)*3600 ) *QCONS          PRNT  45
   65 IF (AMINF.GT.1.09)  GO TO 70                                  PRNT  46
      PSTAG =(PINF + RHOINF*VINF**2 *.5 )/2116.                     PRNT  47
      GO TO 80                                                      PRNT  48
   70 PSTAG = PINF*(1.2*(AMINF**2))**3.5 *(6./(7.*(AMINF**2)-1.))**2.5 PRNT  49
     1   / 2116.                                                    PRNT  50
   80 PS =  PSHK/ 2116.                                             PRNT  51
      WABSEC = WABDOT/DELTA
      WRITE (6,105) TIME,ALT,VINF,ADFA,Q3600,H3600,QN3600,REYN,     PRNT  52
     1              DELTA,WABDOT,WAB,RECI,WALLI,QSAV,QINSD,TRANR,   PRNT  53
     2              HOABL,TABL,PSHK,DYNP,VSHK,SHKI,REFI,QNSUM,SHEARS,PS PRNT  54
     3              ,PSTAG,EMIS,QCONS,WABSEC,NT
C             PSTAG ADDED 1-26-66 REQ.=L.BERGER,CLEARENCE=M.GYETVAY PRNT  56
  105 FORMAT (1H0,7X,5H TIME,16X,3HALT,14X,4HVINF 15X,4HADFA 13X,   PRNT  57
     1 5HQSUBI,13X,5HHSUBI /6E18.8/ 7X,5H QNET,16X,4HREYN 13X,      PRNT  58
     2 5HDELTA 14X,6HWABDOT 13X,3HWAB 15X,4HRECI/ 6E18.8 /7X,       PRNT  59
     3 6H WALLI14X,4HQSAV 14X,5HQINSD 13X,5HTRANR 13X,5HHOABL 13X,4HTABLPRNT  60
     4 /6E18.8/ 7X,5H PSHK14X,4HDYNP14X,4HVSHK14X,4HSHKI14X,4HREFI14X, PRNT  61
     5 5HQNSUM / 6E18.8 /7X,6H SHEAR 13X,2HPS 16X,5HPSTAG 13X,4HEMIS,12XPRNT  62
     6,'QCONS',13X,' WAB/SEC',14X,'NT'/6E18.8,I12)
      IF (NOSB) 106,110,106                                         PRNT  63
  106 RADIN = RADIUS * 12.                                          PRNT  64
      F = VAPRES(100)                                               PRNT  65
      WRITE            (6,107)RADIUS,RADIN ,F,QSONIC,SHEAR,HOABLM   PRNT  66
  107 FORMAT ( 11HORADIUS(FT) 10X,10HRADIUS(IN),10X,1HF,16X,6HQSONIC, PRNT  67
     1        10X,6HSHEAR ,10X,      7H HOABLM / 6E16.8 )           PRNT  68
  110 GO TO (120,140),NDIM                                          PRNT  69
  120 WRITE (6,121) NSEG2,(TEMP(I),I=NSEG2,NSEG)                    PRNT  70
  121 FORMAT( 30H0 1ST SEGMENT PRESENT=NSEG2 =  I2,33H.  TEMP(I), I=NSEGPRNT  71
     12,NSEG FOLLOW.   /  ( 5F12.1 ) )                              PRNT  72
      GO TO 2[illegible]                                            PRNT  73
```

```
  140 WRITF (6,141)                                                   PRNT  74
  141 FORMAT ( 41HO TEMPERATURES FOR 2D FCLLOW                       )PRNT  75
  143 DO 145 I = 1,NAROW                                              PRNT  76
  145 WRITE (6,146) (T2(I,J),J = 1,NBCOL)                             PRNT  77
  146 FORMAT (10F10.1 )                                               PRNT  78
  200 IF (NDBUG.EQ.0) GO TO 30C
      IF (TINF.NF.0.) GO TO 210
      AMINF =0.
      GO TO 215
  210 AMINF =VINF/(49.01*SQRT(TINF))                                  PRNT  80
  215 WRITE (6,COMON)                                                 PRNT 101
  300 IF (NPU.LT.1) GO TO 40C
C            PREVENTS       TSTOR(1)=0.
      IF (TIME.LT.TIME1(1)+DELTA) GO TO 400
      IF (NTMF.LT.1)   GO TO 250
C            PREVENTS DOUBLE 'TSTOR' FOINTS                    8-11-69
      IF (TIME.EQ.TSTOR(NTME))   GO TO 400                            PRNT 103
  250 NTMF = NTME + 1                                                 PRNT 104
      IF (NTMF.LT.51) GO TO  260                                      PRNT 105
      DO 255 I=1,49,2                                                 PRNT 106
      II = (I+1)/2                                                    PRNT 107
      TSTOR(II) = TSTOR(I)                                            PRNT 108
      QO   (II) = QO   (I)                                            PRNT 109
      PO   (II) = PO   (I)                                            PRNT 110
      HSTAB(II) = HSTAB(I)                                            PRNT 111
      HLTAB(II) = HLTAB(I)                                            PRNT 112
  255 UETAB(II) = UETAB(I)                                            PRNT 113
      NTME = II+1                                                     PRNT 114
  260 TSTOR(NTME) = TIME
C             TO PREVENT QO=0, SINCE CHAR DIVIDES BY IT
      IF (Q3600.LE.0.)   Q3600 = 1.E-16                               PRNT 115
      QO   (NTME) = Q3600                                             PRNT 116
      PO   (NTME) = PS                                                PRNT 117
      HSTAB(NTME) = RECI                                              PRNT 118
      HLTAB(NTME) = SHKI                                              PRNT 119
      UETAB(NTME) = VSHK                                              PRNT 120
  400 RETURN                                                          PRNT 121
      END
```

```
      SUBROUTINE PRINTH (ALPHA)                                         FNTRP 34
CPPNH  - SUBROUTINE TO PRINT HOLLERITH (ALPHA)                          FNTRP 33
      WRITE (6,6) ALPHA                                                 FNTRP 35
    6 FORMAT (1H  A6)                                                   FNTRP 36
      RETURN                                                            FNTRP 37
      END                                                               FNTRP 38
```

```
      SUBROUTINE SHOCK                                                  SHCK   3
CSHOCK                                                                  SHCK   1
C - - -      APRIL 8 1964       JOHN KOHLENBERGER         FORTRAN IV    SHCK   2
C     SUBROUTINE SHOCK (ICONF,PINF,TINF,VINF,AOFA,ANGLE,PSHK,TSHK,VSHK) SHCK   4
      DIMENSION    TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)SHCK   5
     1  ,     TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)   SHCK   6
     2  ,     QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)  SHCK   7
     3  ,     A(4,40),B(4,40)                                           SHCK   8
     4  ,     AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)    SHCK   9
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI SHCK  10
     1  ,     RHO, TEMP,TEMPW,TIMF1,TINSD1, VAPRES,VINF1                SHCK  11
     2  ,     AA, BB, CC,CP2, DK, T2                                    SHCK  12
      COMMON       ALT,AMINF,AOFA,ANGLE, CONPR, DELTA,DQSAV             SHCK  13
     1  ,     EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                      SHCK  14
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,           KASE,KFUNT,     NPQSHCK  15
     3  ,     NBCOL,NDBUG,NDIM,NOSB,NPGAS,NQI,NPREST,NPU,NSEG,NSEG1,NSEG2 SHCK  16
     4  ,     NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBT SHCK  17
     5  ,     RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLSHCK  18
     6  ,     SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR  SHCK  19
     7  ,     TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE      SHCK  20
     8  ,     XX,XDIST                                                  SHCK  21
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      SHCK  22
      EQUIVALENCE (NSEG,NAROW)                                          SHCK  23
    5 VSHKS = 0.0                                                       SHCK  24
      ALPHA = (AOFA+ANGLE)/57.2957795                                   SHCK  25
      AMINF = VINF/ (49.01*SQRT(TINF))                                  SHCK  26
      IF (AMINF.LE.1.01)  GO TO 21                                      SHCK  27
   10 GO TO (20,30,120,300,300),ICONF                                   SHCK  28
   20 IF (ALPHA.GT.0.) GO TO 30                                         SHCK  29
   21 PSHK = PINF                                                       SHCK  30
      VSHK = VINF                                                       SHCK  31
      TSHK = TINF                                                       SHCK  32
      GO TO 300                                                         SHCK  33
   30 AMSA = AMINF*SIN (ALPHA)                                          SHCK  34
      IF (AMSA-1.8)40,50,50                                             SHCK  35
   40 PRATIO = -.0661*(AMSA**3)+1.889*(AMSA**2)+.693*AMSA+1.041         SHCK  36
      GO TO 60                                                          SHCK  37
   50 PRATIO = -2.320+4.045*AMSA+1.036*(AMSA**2)+.0161*(AMSA**3)        SHCK  38
   60 PSHK = PRATIO*PINF                                                SHCK  39
      REFM = VINF/1086.                                                 SHCK  40
      RMSA = REFM*SIN (ALPHA)                                           SHCK  41
      IF (RMSA-.6)70,80,80                                              SHCK  42
   70 VPARA = (.1923+1.404*RMSA+1.147*(RMSA**2)+.8325*(RMSA**3))*1.0E+6 SHCK  43
```

```
      GO TO 200
   80 IF (RMSA-6.0)90,100,100                                           SHCK  44
   90 VPARA = (.5958+.4494*RMSA+1.838*(RMSA**2)-.0486*(RMSA**3))*1.0E6  SHCK  45
      GO TO 200                                                         SHCK  46
  100 VPARA = (78.03-19.58*RMSA+3.13*(RMSA**2)-.05054*(RMSA**3))*1.0E6  SHCK  47
  110 GO TO 200                                                         SHCK  48
  113 IF (VSHKS)115,115,118                                             SHCK  49
  115 WRITE (6,350)                                                     SHCK  50
      IGO = 1                                                           SHCK  51
      GO TO 300                                                         SHCK  52
  118 VSHK = SQRT (VSHKS)                                               SHCK  53
      GO TO 300                                                         SHCK  54
C            FOLLOWING 120 IN SHOCK    ***  GRIFFITH - LEWIS  ***       SHCK  55
  120 IF ( (RADIUS.LE.0.).OR.(ALPHA.LE.0.).OR.(XDIST.LE.0.)) GO TO 127  SHCK  56
      Z  = ALPHA*ALPHA*COS(ALPHA)*XDIST/(.802*RADIUS)                   SHCK  57
C          TEST ADDED 8-2-67 FOR OS/360 TO AVOID DIV.ORMULT.BY ZERO     SHCK  58
  126 IF (Z-10.) 300,127,127                                            SHCK  57
  127 AMSA = AMINF*SIN(ALPHA)                                           SHCK  59
  128 IF (AMSA-1.5)130,140,140                                          SHCK  60
  130 PRATIO = 1.007+.3816*AMSA+1.522*(AMSA**2)-.1593*(AMSA**3)         SHCK  61
      GO TO 170                                                         SHCK  62
  140 IF (AMSA-5.0)150,160,160                                          SHCK  63
  150 PRATIO = .2397+1.161*AMSA+1.06*(AMSA**2)+.0489*(AMSA**3)          SHCK  64
      GO TO 170                                                         SHCK  65
  160 PRATIO = -3.182+4.177*AMSA+.8373*(AMSA**2)+.0216*(AMSA**3)        SHCK  66
  170 PSHK = PRATIO*PINF                                                SHCK  67
      REFM = VINF/1086.                                                 SHCK  68
      RMSA = REFM*SIN (ALPHA)                                           SHCK  69
      IF (RMSA-4.0)180,190,190                                          SHCK  70
  180 VPARA = (.157+.04*RMSA+1.91*(RMSA**2)-.107*(RMSA**3))*1.0E+6      SHCK  71
      GO TO 200                                                         SHCK  72
  190 VPARA = (6.137-1.038*RMSA+1.414*(RMSA**2)-.0062*(RMSA**3))*1.0E+6 SHCK  73
  200 VINFS = VINF**2                                                   SHCK  74
      VSHKS = VINFS-VPARA                                               SHCK  75
      IF (VSHKS) 205,205,210                                            SHCK  76
  205 VSHKS = VINFS                                                     SHCK  77
  210 IF (VINF-700.0)     118,118,113                                   SHCK  78
  300 IF(NDBUG-4)   310,305,310                                         SHCK  79
  305 WRITE (6,306) IGO,VINFS,VSHKS,VPARA,PRATIO                        SHCK  80
  306 FORMAT (15H-SHOCK,  IGO=  I3/(6E20.8))                            SHCK  81
  350 FORMAT(16H0 VSHKS NEGATIVE)                                       SHCK  82
  310 RETURN                                                            SHCK  83
      END                                                               SHCK  84
                                                                        SHCK  85
```

```
      SUBROUTINE THERMO                                                 THERM   2
CTHERMO                                                                 THERM   1
C     SUBROUTINE THERMO (IREC,ICONF,TWALL,TINF,PINF,VINF,RHOINF,AOFA,   THERM   3
C    1ANGLE,TSHK,PSHK,RHOSHK,WALLI,RECI,REFMU,REFRHO,REFRM,STAGI)       THERM   4
C - - -     AUG. 18 1964     JOHN KOHLENBERGER          FOR3R+N I5      THERM   5
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)THERM   6
     1  ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)    THERM   7
     2  ,    QI(100),QAD1(100), CP(40),EK(40),EL(40),RHD(40),TEMP(40)   THERM   8
     3  ,    A(4,40),B(4,40)                                            THERM   9
     4  ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)     THERM  10
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI THERM  11
     1  ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                 THERM  12
     2  ,    AA, BB, CC,CP2, DK, T2                                     THERM  13
      COMMON       ALT,AMINF,AOFA,ANGLE, CCNPR, DELTA, DQSAV            THERM  14
     1  ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                       THERM  15
     2  ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,          KASE,KFUNT,    NPQTHERM  16
     3  ,    NBCOL,NDBUG,NDIM,NCSB,NPGAS,NCI,NPPEST,NPU,NSEG,NSEG1,NSEG2 THERM  17
     4  ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  THERM  18
     5  ,    RADIUS,RECI,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLTHERM  19
     6  ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   THERM  20
     7  ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE       THERM  21
     8  ,    XX,XDIST                                                   THERM  22
     9  ,ETATP,ETALP,HVPSTM,PSTMAX                                      THERM  23
      EQUIVALENCE (NSEG,NAROW)                                          THERM  24
      ALPHA = (AOFA+ANGLE)/57.2957795                                   THERM  25
      WALLI=2.2864+.2272*TWALL+.11571E-4*(TWALL**2)+.23676E-9*(TWALL**3)THERM  26
      AIINF = 2.2864+.2272*TINF+.11571E-4*(TINF**2)+.23676E-9*(TINF**3) THERM  27
      AMINF = VINF/(49.01*SQRT (TINF))                                  THERM  28
      IF (NDSB.GT.0) GO TO 212                                          THERM  29
      IF (AMINF.LE.1.01)  GO TO 11                                      THERM  30
      GO TO (10,20,30,210,210),ICONF                                    THERM  31
   10 IF (ALPHA.GT.0.)  GO TO 20                                        THERM  32
   11 TSHK = TINF                                                       THERM   3
      PSHK = PINF                                                       THERM  34
      RHOSHK = RHOINF                                                   THERM  35
      SHKI = 2.2864+.2272*TSHK+.11571E-4*(TSHK**2)+.23676E-9*(TSHK**3)  THERM  36
C           FOR ASCENT TRAJECTORY . . . 11/4/68    PETE CROWL    JMK
      VSHK = VINF
      SHKMU = .42642E-7*(SHKI**.493)
C           FOR ASCENT TRAJECTORY . . . 3/14/69    MIKE GYETVAY   JMK
      GO TO 40                                                          THERM  37
   20 AMSA = AMINF*SIN (ALPHA)                                          THERM  38
      IF (AMSA-8.0)25,25,28                                             THERM  39
```

```
   25 SHKI = AIINF*(.9167+.3203*AMSA+.224*(AMSA**2)-.4484E-3*(AMSA**3))  THERM 40
      GO TO 40                                                          THERM 41
   28 SHKI = AIINF*(1.107-.2209*AMSA+.3644*(AMSA**2)-.008462*(AMSA**3))  THERM 42
      GO TO 40                                                          THERM 43
C           IN THERMO, BELOW 30      ***  GRIFFITH - LEWIS               THERM 44
   30 IF ((RADIUS.LE.0.).OR.(ALPHA.LE.0.).OR.(XDIST.LE.0.)) GO TO 31     THERM 45
      Z =  ALPHA*ALPHA*COS(ALPHA)*XDIST/(.892*RADIUS)                   THERM 46
C           TEST ADDED 8-2-67 FOR OS/360 TO AVOID DIVIDING BY ZERO      THERM 45
  298 IF (Z-10.) 290,31,31                                              THERM 47
  290 IF (Z-1.5) 300,300,295                                            THERM 48
  295 CSUBP = 2.46 * ALPHA**2                                           THERM 49
      GO TO 310                                                         THERM 50
  300 CSUBP = 2.*ALPHA**2*(((((-6.3*Z+29.3)*Z-52.2)*Z+43.15)*Z-15.5)*Z   THERM 51
     1  +2.67)                                                          THERM 52
  310 PSHK = PINF*(.7*AMINF**2*CSUBP+1.)                                THERM 53
      VSHK = SQRT(VINF*VINF*(1.-(PSHK/PINF)**.286 *.8333/(AMINF**2)     THERM 54
     1      *(1.168*(AMINF**2)- .1663)**.714))                          THERM 55
      STAGI=AIINF+(VINF**2)/50061.5                                     THERM 56
      SHKI = STAGI - VSHK*VSHK/50061.5                                  THERM 57
      GO TO 40                                                          THERM 58
   31 AMSA = AMINF*SIN(ALPHA)                                           THERM 59
      IF (AMSA-8.0)35,35,38                                             THERM 60
   35 SHKI = AIINF*(1.03+.0827*AMSA+.2174*(AMSA**2)-.6956E-3*(AMSA**3))  THERM 61
      GO TO 40                                                          THERM 62
   38 SHKI = AIINF*(1.106-.3695*AMSA+.3466*(AMSA**2)-.007766*(AMSA**3))  THERM 63
      GO TO 40                                                          THERM 64
   40 PR = .76                                                          THERM 65
      GO TO (50,60,120),IREC                                            THERM 66
   50 REC = SQRT(PR)                                                    THERM 67
      GO TO 70                                                          THERM 68
   60 REC = PR**(.33333)                                                THERM 69
C           ECKERTS REFERENCE ENTHALPY EQUNS.                           THERM 70
   70 RECI = SHKI+(REC*(VSHK**2))/50061.5                               THERM 71
      REFI = SHKI+0.5*(WALLI-SHKI)+.22*(RECI-SHKI)                      THERM 72
      IF (REFI-1300.)80,90,90                                           THERM 73
   80 REFMU = .42642E-7*(REFI**.493)                                    THERM 74
      REFRHO = .576E-4*(REFI**(-.849))*PSHK                             THERM 75
      GO TO 100                                                         THERM 76
   90 REFMU = .28428E-6*(REFI**.228)                                    THERM 77
      REFRHO = .865E-5*(REFI**(-.584))*PSHK                             THERM 78
  100 REFPM = .2465E-11*(REFI**(-.356))*PSHK                            THERM 79
  110 STAGI = AIINF+(VINF**2)/50061.5                                   THERM 80
      GO TO 200                                                         THERM 81
C           SONIC POINT (WEDGE OR CONE)                                 THERM 82
  120 IF (SHKI-1300.)130,140,140                                        THERM 83
  130 SHKMU = .42642E-7*(SHKI**.493)                                    THERM 84
```

```
      RHOSHK = .576E-4*(SHKI**(-.849))*PSHK                             THERM 85
      GO TO 200                                                         THERM 86
  140 SHKMU = .28428E-6*(SHKI**.228)                                    THERM 87
      RHOSHK = .865E-5*(SHKI**(-.584))*PSHK                             THERM 88
  200 IF (NDBUG-5) 205,201,205                                          THERM 89
  201 WRITE (6,203) AIINF,REFI,RECI,REC,  ALPHA,AMSA,WALLI,TWALL        THERM 90
  203 FORMAT (14H- THERMO DEBUG / 5X,5HAIINF 15X,4HREFI 16X,4HRECI 16X, THERM 91
     1       3HREC / 4E20.8 / 5X,5HALPHA 15X,4HAMSA 16X,5HWALLI 15X,    THERM 92
     2       5HTWALL / 4E20.8 )                                         THERM 93
  205 RETURN                                                            THERM 94
  210 IF (ICHK.GT.0) GO TO 212                                          THERM 95
  211 PSHK = PINF*(1.20*AMINF**2)**3.5*(6./(7.*AMINF**2-1.))**2.5       THERM 96
C            C. ANDERSON  MOD.           03/15/65                       THERM 97
  212 STAGI = AIINF+(VINF**2)/50061.5                                   THERM 98
      RECI = STAGI                                                      THERM 99
C           C.B.A.   8-19-64  CHANGE                                    THERM100
      GO TO 200                                                         THERM101
      END                                                               THERM102
```

```
      SUBROUTINE TRAJ                                                   ALTM   2
CTRAJ                                                                   ALTM   1
C - - -     APRIL 22 1964       JOHN KOHLENBERGER         FORTRAN IV    ALTM   3
C     INCREASES TIME BY DELTA THEN LINEARLY INTERPOLATES TO GET ALT,    ALTM   4
C     VINF, AOFA, HINSD, TINSD, AND QCONS AS FUNCTIONS OF TIME          ALTM   5
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)ALTM   6
     1  ,   TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)     ALTM   7
     2  ,   QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)    ALTM   8
     3  ,   A(4,40),B(4,40)                                             ALTM   9
     4  ,   AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)      ALTM  10
      COMMON       A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI ALTM  11
     1  ,   RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                  ALTM  12
     2  ,   AA, BB, CC,CP2, DK, T2                                      ALTM  13
      COMMON       ALT,AMINF,AOFA,ANGLE, CCNPR, DELTA,DQSAV             ALTM  14
     1  ,   EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                        ALTM  15
     2  ,ICHK,ICK,ICONE,IGO,INB4,INB42,IREC,           KASE,KFUNT,     NPOALTM  16
     3  ,   NBCOL,NDBUG,NDIM,NOSB,NPGAS,NCI,NPREST,NPU,NSEG,NSEG1,NSEG2 ALTM  17
     4  ,   NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI   ALTM  18
     5  ,   RADIUS,REC1,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLALTM  19
     6  ,   SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR    ALTM  20
     7  ,   TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE        ALTM  21
     8  ,   XX,XDIST                                                    ALTM  22
     9     ,ETATP,ETALP,HVPSTM,PSTMX,NT
      COMMON / SWALOW/  QINT(50),HEDGE(50),PEDGE(50),UEDGE(50),HRECOV(50ALTM  24
     1)  ,  INPCW                                                       ALTM  25
C           FOR ARCHIE OSSIN - COLD WALL INPUTS FROM ENTROPY SWALLOWING ALTM  26
      EQUIVALENCE (NSEG,NARCW)                                          ALTM  27
   10 TIME = TIME+DELTA                                                 ALTM  29
      NT = NT + 1
   15 IF(NTRAJ-2)105,55,20                                              ALTM  30
   20 DO 30 I=1,NTRAJ                                                   ALTM  31
   25 IF(TIME-TIME1(I))50,110,30                                        ALTM  32
   30 CONTINUE                                                          ALTM  33
   40 I = NTRAJ                                                         ALTM  34
   50 IF(I-1)55,55,60                                                   ALTM  35
   55 I = 2                                                             ALTM  36
   60 FACTOR = (TIME-TIME1(I-1))/(TIME1(I)-TIME1(I-1))                  ALTM  37
   70 ALT = (ALT1(I)-ALT1(I-1))*FACTOR + ALT1(I-1)                      ALTM  38
      VINF = (VINF1(I)-VINF1(I-1))*FACTOR + VINF1(I-1)                  ALTM  39
   80 AOFA=(AOFA1(I)-AOFA1(I-1))*FACTOR+AOFA1(I-1)                      ALTM  40
      HINSD = (HINSD1(I)-HINSD1(I-1))*FACTOR + HINSD1(I-1)              ALTM  41
   90 TINSD = (TINSD1(I)-TINSD1(I-1))*FACTOR + TINSD1(I-1)              ALTM  42
      QCONS = (QCONS1(I)-QCONS1(I-1))*FACTOR + QCONS1(I-1)              ALTM  43
```

```
      IF (INPCW.LE.0) GO TO 100                                          ALTM  44
      QSUBI =(QINT(I-1)+  FACTOR*(QINT(I)-QINT(I-1)))*3600.              ALTM  45
      SHKI  = HEDGE(I-1) +FACTOR*(HEDGF (I)-HEDGF (I-1                   ALTM  46
      VSHK  = UEDGE(I-1) +FACTOR*(UEDGE (I)-UEDGE (I-1))                 ALTM  47
      PSHK  =(PEDGE(I-1) +FACTOR*(PEDGE (I)-PEDGE (I-1))*2116.8          ALTM  48
      RECI  = HRECOV(I-1)+FACTOR*(HRECOV(I)-HRECOV(I-1))                 ALTM  49
      IF (INPCW.NE.NTRAJ) INPCW = NTRAJ                                  ALTM  50
  100 IF((VINF.LE.0.).AND.(INPCW.GT.0))   VINF = VSHK                    TRAJ  52
      IF (NDBUG.NE.3)               RETURN                                     52I
      WRITE (6,101) ALT,VINF,AOFA,HINSD,TINSD,QCONS                            52H
  101 FORMAT(' TRAJ-    ALT',13X,'VINF',12X,'AOFA',12X,'HINSD',11X,            52G
     1  'TINSD',11X,'QCONS' / 6E16.8)                                          52F
      IF (INPCW.LE.0)               RETURN                                     52E
      WRITE (6,102) QSUBI,SHKI,VSHK,PSHK,RECI                                  52D
  102 FORMAT (7X,'QSUBI',11X,'SHKI',12X,'VSHK',12X,'PSHK',12X,'RECI' /         52C
     1  5E16.8)                                                                52B
      RETURN                                                                   52A
  105 I=1                                                                ALTM  53
  110 ALT = ALT1(I)                                                      ALTM  54
      VINF = VINF1(I)                                                    ALTM  55
  120 AOFA = AOFA1(I)                                                    ALTM  56
      HINSD = HINSD1(I)                                                  ALTM  57
  130 TINSD = TINSD1(I)                                                  ALTM  58
      QCONS = QCONS1(I)                                                  ALTM  59
      IF (INPCW.LE.0) GO TO 100                                          ALTM  60
      QSUBI = QINT  (I)   *  3600.                                       ALTM  61
      SHKI  = HEDGE (I)                                                  ALTM  62
      VSHK  = UEDGE (I)                                                  ALTM  63
      PSHK  = PEDGE (I) *2116.8                                          ALTM  64
      RECI  = HRECOV(I)                                                  ALTM  65
      IF (INPCW.NE.NTRAJ) INPCW = NTRAJ                                  ALTM  67
  140 GO TO 100                                                          ALTM  68
C     MAYBE AT 100 CONVERT AOFA TO RADIANS                               ALTM  69
      END                                                                ALTM  70
```

```
      SUBROUTINE TWOD                                                   TEMM   3
CTWOD                                                                   TEMM   1
C - - -    MARCH 27 1969      JOHN KOHLENBERGER          FORTRAN IV    TEMM   3
      DIMENSION   TIME1(100),ALT1(100),VINF1(100),AOFA1(100),HINSD1(100)TEMM   4
     1   ,    TINSD1(100),QCONS1(100),VAPRES(100),CPG(100),TEMPW(100)   TEMM   5
     2   ,    QI(100),QAD1(100), CP(40),EK(40),EL(40),RHO(40),TEMP(40)  TEMM   6
     3   ,    A(4,40),B(4,40)                                           TEMM   7
     4   ,    AA(10),BB(10),CC(10,10),CP2(10,10),DK(10,10),T2(10,10)    TEMM   8
      COMMON      A,ALT1,AOFA1,B, CP,CPG, EK,EL, HINSD1, QAD1,QCONS1,QI TEMM   9
     1   ,    RHO, TEMP,TEMPW,TIME1,TINSD1, VAPRES,VINF1                TEMM  10
     2   ,    AA, BB, CC,CP2, DK, T2                                    TEMM  11
      COMMON        ALT,AMINF,AOFA,ANGLE, CONPR, DELTA, DQSAV           TEMM  12
     1   ,    EMIS,ETA1,ETA2, HINSD,HOABL,HSUBI,HV                      TEMM  13
     2   ,ICHK,ICK,ICONF,IGO,INB4,INB42,IREC,          KASE,KFUNT,     NPQTEMM  14
     3   ,    NBCOL,NDBUG,NDIM,NOSB,NPGAS,NCI,NPREST,NPU,NSEG,NSEG1,NSEG2 TEMM  15
     4   ,    NTRAJ, PINF,PR,PSHK,PSL,QCALT,QCONS,QINSD,QNET,QSAV,QSUBI  TEMM  16
     5   ,    RADIUS,REC1,REFI,REFMU,REFRHO,REFRM,REYN,RHOINF,RHOSHK,RHOSLTEMM  17
     6   ,    SHKI,SHKMU, STOP,SWEEP,STAGI, TABL,TIME,TINF,TINSD,TRANR   TEMM  18
     7   ,    TSHK,TWALL,TWNB, VINF,VSHK, W,WAB,WABDOT,WALLI,WRITE      TEMM  19
     8   ,    XX,XDIST                                                  TEMM  20
     9   ,ETATP,ETALP,HVPSTM,PSTMAX                                     TEMM  21
      EQUIVALENCE (NSEC,NAROW)                                          TEMM  22
      COMMON /MIKE/ ADAM(64),JESHUA(2),ENCCH(4),ISTETS,NOCHEK
      DIMENSION T2NEW(10,10)                                            TEMM  23
C     2D          1D                                                    TEMM  24
C             MATERIALS MUST BE HOMOGENEOUS IN X (BUT NOT Y) DIRECTION  TEMM  25
C     AA(I)  = SEGMENT THICKNESS (Y DIRECTION)- INCHES     = EL(I)      TEMM  26
C     BB(K)  = SEGMENT LENGTH (X DIRECTION)   - INCHES                  TEMM  27
C     CC(I,K) =1 FOR CONDUCTION  INTO AND OUT-OF  NODAL POINT I,K       TEMM  28
C              =0 FOR NO HEAT CONDUCTION TO OR FROM NODAL POINT (I,K)   TEMM  29
C     CP2(I,K) = CP(I)                                                  TEMM  30
C     DK(I,K) = EK(I)                                                   TEMM  31
C     T2(I,K) = TEMP(I)                                                 TEMM  32
    5 CALL GECPK2                                                       TEMM  33
C     STABMX = 0.
   10 DO 125 I = 1,NAROW                                                TEMM  34
   30 DO 125 K = 1,NBCOL                                                TEMM  35
   50 IF (CC(I,K) .GT.0.)    GO TO 60
   55 T2NEW(I,K) = 0.0                                                  TEMM  37
      GO TO 125                                                         TEMM  38
   60 FACTOR = DELTA/(25.0*RHO(I)*CP2(I,K)*BB(K)*AA(I))                 TEMM  39
      IF (I.GT.1)   GO TO 70
CHANGED     3-4-69   FOR RJGILBERT
```

```
CHANGED       3-27-69   FOR R.GILBERT
   65 IF (K.LE.1)    GO TO 66
      IF((T2(1,K-1)-T2(1,K))**2.LE.100.)GO TO 66
      TWALL = T2(1,K)
      CALL ENCO
   66 TERM1 = QNET*BB(K)/12.0                                          TEMM  41
CHANGED       3-27-69   FOR R.GILBERT
      GO TO 75                                                          TEMM  42
   70 TERM1 = BB(K)*CC(I-1,K)/(.5*(AA(I-1)/DK(I-1,K)+AA(I)/DK(I,K)))    TEMM  43
     1*(T2(I-1,K)-T2(I,K))                                              TEMM  44
   75 IF (I.GE.NAROW)    GO TO 85
   80 TERM2 = BB(K)*CC(I+1,K)/(.5*(AA(I+1)/DK(I+1,K)+AA(I)/DK(I,K)))    TEMM  46
     1*(T2(I+1,K)-T2(I,K))                                              TEMM  47
      GO TO 90                                                          TEMM  48
   85 TERM2 = HINSD*(TINSD-T2(I,K))/12.0*BB(K)                          TEMM  49
   90 IF (K.GT.1)    GO TO 100
   95 TERM3 = 0.0                                                       TEMM  51
      GO TO 105                                                         TEMM  52
  100 TERM3 = AA(I)*CC(I,K-1)/(.5*(BB(K-1)/DK(I,K-1)+BB(K)/DK(I,K)      TEMM  53
     1))*(T2(I,K-1)-T2(I,K))                                            TEMM  54
  105 IF (K.LT.NBCOL)    GO TO 115
  110 TERM4 = 0.0                                                       TEMM  56
      GO TO 120                                                         TEMM  57
  115 TERM4 = AA(I)*CC(I,K+1)/(.5*(BB(K+1)/DK(I,K+1)+BB(K)/DK(I,K)      TEMM  58
     1))*(T2(I,K+1)-T2(I,K))                                            TEMM  59
  120 T2NEW(I,K) =  FACTOR *(TERM1+TERM2+TERM3+TERM4)                   TEMM  60
      IF (NDBUG.GE.14) WRITE(6,1200) TIME,T2NEW(I,K),FACTOR,TERM1,TERM2,
     1TERM3,TERM4,AA(I),BB(K),CP2(I,K),RHC(I),DK(I,K),QNET,I,K,STAB
 1200 FORMAT (' DEBUG- TWOD - TIME='E16.8,' T2NEW='E16.8,' FACTOR='E16.8
     1/ 7X,'TERM1',7X,'TERM2',7X,'TERM3',7X,'TERM4',7X,'AA',10X,'BB',10X
     2 ,'CP2', 9X,'RHO',10X,'DK'/9E12.4/
     3                         ' QNET=',E16.8,'I=',I6,'K=',I6,' STAB=',E13.5)
      IF (I.GT.1)    GO TO 121
      DTL = 0.0
      GO TO 1210
  121 DTL =  T2(I-1,K)-T2(I,K)                                          TEMM 610
 1210 IF (I.LT.NAROW) GO TO 122
      DTR = 0.0
      GO TO 1220
  122 DTR =  T2(I+1,K)-T2(I,K)                                          TEMM 611
 1220 IF (K.GT.1)    GO TO 123
      DTD = 0.0
      GO TO 1230
  123 DTD =  T2(I,K-1)-T2(I,K)                                          TEMM 612
 1230 IF (K.LT.NBCOL) GO TO 124
      DTU = 0.0
```

```
      GO TO 1240
  124 DTU =  T2(I,K+1)-T2(I,K)                                         TEMM 613
 1240 CONTINUE
      IF (DTL.EQ.0.) GO TO1121                                          TEMM 614
      TERM1 = TERM1/DTL                                                 TEMM 615
 1121 IF (DTR.EQ.0.) GO TO1122                                          TEMM 616
      TERM2 = TERM2/DTR                                                 TEMM 617
 1122 IF (DTD.EQ.0.) GO TO1123                                          TEMM 618
      TERM3 = TERM3/DTD                                                 TEMM 619
 1123 IF (DTU.EQ.0.) GO TO1124                                          TEMM 620
      TERM4 = TERM4/DTU                                                 TEMM 621
 1124 STAB = FACTOR *(TERM1+TERM2+TERM3+TERM4)                          TEMM 622
C     STABMX = AMAX (STAB,STABMX)
      IF (STAB.LE.1.) GO TO 125                                         TEMM  63
      IF (NOCHEK.GT.1)   GO TO 125
                          WRITE(6,1200) TIME,T2NEW(I,K),FACTOR,TERM1,TERM2,
     1TERM3,TERM4,AA(I),BB(K),CP2(I,K),RHC(I),DK(I,K),QNET,I,K,STAB
      WRITE (6,126)     DTL,DTP,DTD,DTU,  T2(I-1,K),T2(I+1,K),
     1        T2(I,K-1),T2(I,K+1),        HINSD,TINSD
  126 FORMAT (20X,'LEFT',12X,'RIGHT',11X,'DOWN',12X,'UP'/ ' D(T2)',4F16.
     1 8/' T2   ',4E16.8/ 10X,           ' HINSD=',E13.5,' ,TINSD=',E13.5
     2)
      WRITE (6,127)
  127 FORMAT (1H-,20X,'  STABILITY ERROR .... AT ABOVE CONDITIONS'/1H-)
  125 CONTINUE                                                          TEMM  67
C            SHOULD CUT TIME STEP (DELTA) HERE  IF STABMX.GT.1.
C      OR  INCREASE DELTA  IF STABMX.LT. .5
      DO 150 I = 1,NAROW                                                TEMM  68
      DO 150 K = 1,NBCOL                                                TEMM  69
  150 T2(I,K) = T2NEW(I,K)  +T2(I,K)                                    TEMM  70
  160 TWALL = T2(1,1)                                                   TEMM  71
      RETURN                                                            TEMM  72
C  50 IF (CC(I,K))55,55,60                                              TEMM  36
C     IF (I-1)65,65,70                                                  TEMM  40
C  75 IF (I-NAROW)80,85,85                                              TEMM  45
C  90 IF (K-1)95,95,100                                                 TEMM  50
C 105 IF (K-NBCOL)115,110,110                                           TEMM  55
      END                                                               TEMM  73
```

```
      SUBROUTINE ZERBLK
COMMON /      /  INITIALIZE
C            TO INITIALIZE BLANK COMMON
C - - -        APR      1969
      COMMON A(2154)   ,ICHK(22),PINF(46)
      DIMENSION ALTIO(2154),ALT2(46),IZZAK(22)
C     EQUIVALENCE (ALTIO(1),A(1)  ),(PINF(1),ALT2(1)),(ICHK(1),IZZAK(1))
      DATA ALTIO/1740*0.,100*1.,100*10.,100*.01,114*0./
     1 ,                              IZZAK/0,1,3,-1,0,0,3,5*0,1,5*0,1,0,0,0/
     2 ,ALT2/ 3*0.,760.,15*0.,.0023796,19*0.,1.,6*0./
      NAMELIST /BLANK/  ALTIO,IZZAK,ALT2
      DO 422 I=1,2154
      IF (I.GT.46)   GO TO 422
      PINF(I) = ALT2(I)
      IF (I.GT.22)   GO TO 422
      ICHK(I) = IZZAK(I)
  422 A(I) = ALTIO(I)
      IF (A     (1).EQ.A     (1078))    GO TO 7
      WRITE (6,6)
    6 FORMAT ('    BLANK COMMON NOT PROPERLY INITIALIZED')
      WRITE (6,BLANK)
    7 RETURN
      END
```

```
C          END OF ASAH PROGRAM DECK
C
C
C          BEGINNING OF ASAH SAMPLE DATA
 SAMPLE 1 AND 2 DIMENSIONAL PROBLEMS - FOR A.S.A.H. MAR.3,'71
 &BD060S
 TITLE=' SAMPLE ONE DIMENSIONAL PROBLEM, ONE SEGMENT, AERO HEAT',
 ISTETS=0,AOFA=16*0,HVPSTM=0.,PSTMAX=0.,KASE=1,NDIM=1,
 ICONF=3,ANGLE=8.,DELTA=.05,WRITE=10.,NSEG=1,KFUNT=4, NTRAJ=16
 EMIS=.85,TABL=10000.,HV=1000.,ETAL=1.,ETAT=1.,
 TIME=0.,22.,40.,54.,60.,70.,80.,96.,112.,118.,124.,130.,136.,
 140.,144.,148.,
 ALT=200.,4350.,15500.,19540.,37000.,51570.,69210.,104800.,151270.,
 172125.,195210.,220540.,245180.,261700.,277990.,294230.,
 VINF=10.,410.,915.,1410.,1690.,1850.,2000.,2630.,4000.,5850.,7550.,
     8690.,9350.,9750.,9970.,10110.,10270.,
 A1=60,    B1=.28,     RHO=108.,   EL=.125,     STOP=14.8,  XDIST=.243,
 &END           ONE DIM., ONE SLAB, ONE MAT'L
 &BD060S  KASE=2,TITLE(9)=' - W','ITH ','COLD',' WAL','L IN','PUTS',
 ', 5 ','SEG.',
 KFUNT=3,
 XDIST=.243,TRANLT=80000.,NTRAJ=17,INPCW=17,
 TIME=0.,22.,40.,54.,60.,67.,70.,80.,96.,112.,118.,124.,130.,136.,140.,
      144.,148.,
 QINT=.1,.28,1.00,15.9,15.6,11.36,6.42,3.97,2.88,3.43,4.12,3.68,
     2.57,1.57,1.08,.711,.375,
 ALT=200.,4350.,15500.,19540.,37000.,47000.,51570.,69210.,104800.,
      151300.,172125.,195210.,220540.,245180.,261700.,277990.,295000.,
 PEDGE=.995,.84,.55,.35,.25,.15,.13,.062,.010,.0033,.00131,6.6E-4,
      2.65E-4,9.47E-5,4.23E-5,1.86E-5,7.69E-6,
 HRECOV=123.,123.,126.,140.,146.,160.,167.,221.,400.,700.,1093.,1393.,
      1574.,1683.,1753.,1800.,1907.,
 UEDGE=10.,400.,900.,1300.,1550.,1775.,1800.,2400.,3600.,5500.,7149.,
      8320.,9084.,9593.,9802.,9977.,10000.,
 HEDGE=123.,123.,120.,130.,165.,165.,165.,170.,190.,210.,242.,240.,
      199.,149.,153.,140.,127.,
 TEMP=8*540., TRANLT=47000.,
 NSEG=5,
 A1=60.,.25,.965,2*75.,
 B1=.0,.28,.125,2*.034,B2=.0003,4*.0,
 RHO=108.,150.,334.,2*1192.,
 EL=.125,.010,.050,2*.25,
 WRITE=5.,
```

```
&END      NOTE - VARIABLE THERMAL PROPERTIES
&BD060S QCONS=5*.495,
&END
&BD060S  TITLE(15)=', SI','X SE','GMEN','TS  ',   KASE=3,
NSEG=6,
QCONS=6*1.,
TEMP=6*540.,
A1=.14,60.,.25,.965,2*75.,
B1=.26,.0,.28,.125,2*.034, B2=.0,.0003,4*.0,
RHO=136.,108.,150.,334.,2*1192.,
EL=.010,.125,.010,.050,2*.25,
EMIS=.15,HV=950. ,ETAL=.43,ETAT=.17,TABL=1800.,
&END      NOTE - CHANGED EMIS, HV, TABL, ECT.
&BD060S  TITLE=' TWO-DIMENSIONAL PROBLEM, BASED ON KASE 3 (1-D)'
KASE=23, NDIM=2,
WRITE=.02,
START= 68.7,
&END
&TWODIM
AA=.125,.010,.050,2*.25,
BB=.308,.308,
CC=5*1,5*0,1,0,3*1,80*0.,
T2=5*540.,5*0,540,0,3*540,85*0,
NAROW=5,
NBCOL=2,
T2=657,632,577,546,545,5*0,665,0,3*544,85*0,
&END
ENDDATA
&BD060S  TITLE=' SAMPLE PROBLEM  OUTPUT  (SECTION 6 )
NTRAJ=9,
ICONF =3,  ANGLE=10,   DELTA=.01,  WRITE=1,    STOP=9.0,
EMIS=.55,  TABL=4900,  HV=5500,    ETAL=.52,   ETAL=.25,   RHO=20*102,
A1=20.*.29,B1=20*.26,  TEMP=20*560,EL=20*.2,   XDIST=.16667
NOSB=1,    RADIUS=.0833,     TWALL=4500,
XI=.75,    HF=60,
           HV2=1000,   ETAT2=.25,  ETAL2=.52,
NOPSTG=6,  NOPSHR=6,
STGTBL=  0,13.66,   20,   30,   44,  200,
FV1S=   .69,  .50,  .34,  .20,   2*.125,
FV2S=   6*.55,
SHRTBL=  0,  150,  220,  280,  310,  500,
FV1P=      2*.85,   .74,  .49,   2*.125,
FV2P=   6*.55
TIME=0,    1.0,    2.0,    3.0,    4.0,   5.0,  6.0,    7.0,    9.0,
ALT =299000,291470,283929,276378,268817,261246,253665,246075,230870,
VINF=22000, 22009.,22018, 22026, 22035, 22041, 22047, 22052, 22057,
```

```
3047      &END
3048      ENDDATA              SAMPLE INPUT FOR THIS PROBLEM IS IN SECTION 5
3049      ENDDATA
3050      RHO=108,EL=.125,NPU=0,NSEG=1,STOP=148,KFUNT=3,
3051    C          END OF TAPE 06932  (MADE APRIL 15, 1971)
END OF FILE READ ON UNIT 11.
.DONE                                                            BLK1 314
```